Bring the Content Closer to the End User
In-Network Adaptation and Caching of Mobile Video

Ghoreishi, Seyed Ehsan

Awarding institution:
King's College London

Bring the Content Closer to the End User: In-Network Adaptation and Caching of Mobile Video

Seyed Ehsan Ghoreishi

A Thesis Submitted for the Degree of

Doctor of Philosophy at

King's College London

April 2017

Acknowledgments

I would like to express my special appreciation and thanks to my supervisor Professor Hamid Aghvami, you have been a great mentor for me. I would like to thank you for your patient guidance and continuous encouragement for the completion of this research work.

I would also like to thank Professor Mischa Dohler, Dr Vasilis Friderikos, Dr Nishanth Sastry and Dr Mohammad Shikh-Bahaei for their important contributions and valuable advice, knowledge and many insightful discussions and suggestions. I extend my special appreciation to Dr Dmytro Karamshuk for offering his eager support and contribution on many technical matters.

I am very much thankful to my examiners Professor Izzat Darwazeh and Professor Mohammad Ghanbari for their deep and constructive comments and suggestions, which have led to significant improvements in the quality of this thesis.

I am grateful to the students and staff at the Centre for Telecommunications Research (CTR), in particular Adnan, Aravindh, Bright, Christoforos, Fahad, Frank, Gao, Giorgos, Hisham, Omar, Sagar, Sobhan, Sumayyah, Shuyu, Syed, Maria, Massimo, Yansha, Yaqub and Nikki.

I would like to express my special thanks to my mother and father. Words cannot express how grateful I am for all of the sacrifices that you have made on my behalf. I owe all the success in life to you.

Abstract

The extensive growth in smartphone and tablet market has led to a continuous increase in mobile video traffic, which has urged mobile network operators (MNOs) to redesign their networks and search for cost-effective solutions to bring content closer to the end user. This enables support for more simultaneous video streams, while maintaining stringent delay bounds. However, with adaptive bit rate streaming (ABS) which provides multiple source video bit rates for a single video to meet the heterogeneity of user devices and network conditions, caching all rate variants significantly increases backhaul and storage requirements. Therefore, having cached the highest quality of video contents, this thesis proposes two methodologies to perform in-network video adaptation: (1) a perceptual quality-aware video adaptation scheme that encodes video sequences at a target bit rate; (2) a quality of experience (QoE)-aware video adaptation technique which drops packets from scalable video streams to produce lower bit-rate versions under QoE and delay constraints. These adaptation schemes then allocate resources to meet the delay limitation of the lower rate streams for power-efficient streaming over downlink OFDMA systems.

Alternatively, instead of transrating reactively cached contents, operators can implement intelligent caching in their networks. Predictable user demands can then be proactively served from content caches deployed at mobile gateways in the vicinity of users. Therefore, this thesis also evaluates the potential benefits from in-network caching of scalable videos and finds the trade-off between the

potential savings from- and infrastructural costs of in-network caching.

In light of the increasing trend in virtualization of network functions, a cost-effective Caching-as-a-Service (CaaS) framework for virtual video caching in 5G mobile networks is proposed in this study. In order to evaluate the pros and cons of this CaaS approach, a virtual caching problem is formulated in order to maximize return on caching investment by finding the best trade-off between the cost of cache storage and bandwidth savings from caching video contents in the MNO's cloud.

Contents

List of Figures

List of Figures

List of Tables

Acronyms

ABS adaptive bit rate streaming.

APA adaptive power allocation.

API application programming interface.

AVC advanced video coding.

AWGN additive white Gaussian noise.

BBU baseband unit.

BER bit-error rate.

BHR byte hit ratio.

BIP binary-integer programming.

BS base station.

CaaS Caching-as-a-Service.

CAPEX capital expenditures.

CDF cumulative distribution function.

CDN content delivery network.

CDT canonical duality theory.

CIF common intermediate format.

CN core network.

CNR channel-to-noise ratio.

CP content provider.

CSI channel state information.

DASH dynamic adaptive streaming over HTTP.

DSA dynamic subcarrier assignment.

EPC evolved packet core.

EPCaaS evolved packet core (EPC) as a Service.

GD-Size greedy dual size.

GoP group of pictures.

HEVC high efficiency video coding.

HR hit ratio.

ISP Internet service providers.

IWO Invasive Weed Optimization.

JSVM joint scalable video model.

JVT Joint Video Team.

KKT Karush-Kuhn-Tucker.

LFU least frequently used.

LRU least recently used.

LTE 3GPP long term evolution.

MNO mobile network operator.

MOT maximum offloaded traffic.

MPD media presentation description.

MRI maximum return on investment.

MS-SSIM multi-scale structural similarity.

NAT network address translator.

OFDMA Orthogonal Frequency Division Multiple Access.

OPEX operating expenditures.

P-GW packet data network gateway.

PSNR peak signal-to-noise-ratio.

QoE quality of experience.

QoS quality of service.

RA resource allocation.

RAN radio access network.

RANaaS radio access network (RAN) as a Service.

RB resource block.

RR random replacement.

RRH remote radio head.

RTP Real-time Transport Protocol.

RTT round trip time.

S-GW serving gateway.

SLA service-level agreement.

SNR signal to noise ration.

SP service provider.

SVC scalable video coding.

TTI transmission time interval.

VM virtual machine.

VOD video on demand.

VQ virtual queue.

Chapter 1

Introduction

The growing popularity of mobile video services has led to considerably increased volumes of network traffic. According to the recent reports [4], mobile video will represent 75% of global mobile data traffic by 2020. Therefore, mobile operators are searching for cost-effective ways to bring content closer to the end user, which increases the network's capacity to serve more simultaneous video streams while maintaining stringent delay bounds.

One approach lies in placing geographically distributed content delivery networks (CDNs). CDNs allow for high-performance delivery of content with many dispersed components work jointly to distribute the load among servers that are close to the users (see Fig. 1.1) [5]. However, in order to reach an end user's device, CDN-served traffic must still cross through the mobile operator's core network (CN) and RAN. The significant strain on the operator's CN and RAN backhaul contributes to congestion, delays in streaming video content. Additionally, it puts a constraint on the network's capacity to serve a large number simultaneous video requests.

Many studies have proposed in-network video caching as a way to maximize the video capacity of wireless networks, while enhancing the user-perceived quality of experience (QoE) [6–9]. With in-network caching, users can access popu-

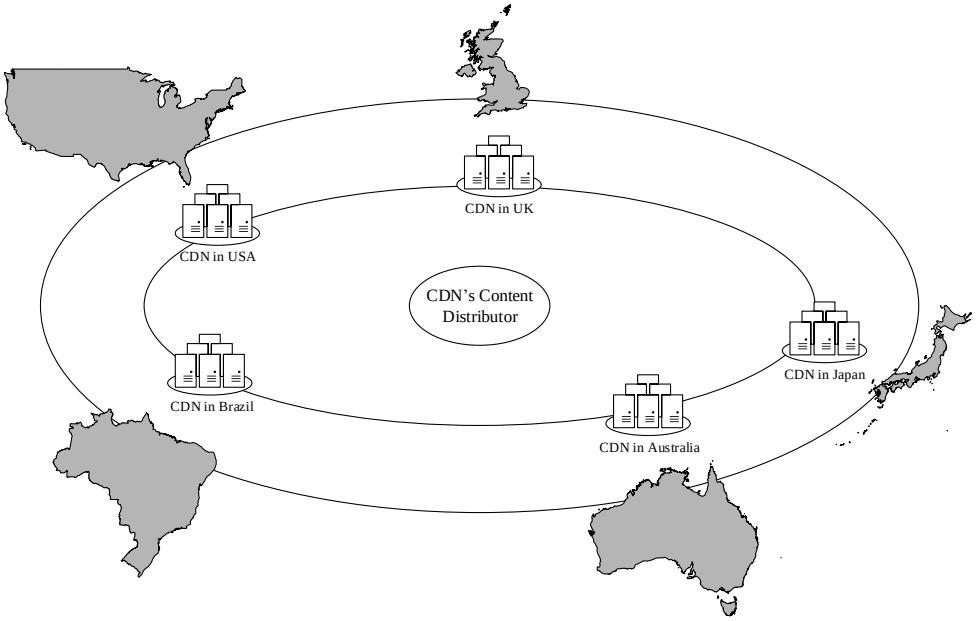


Fig. 1.1: Common architecture for Internet video (adapted from [1])

lar content from caches of nearby mobile network operator (MNO) gateways, i.e. EPC and RAN nodes [10–15], hence significantly reducing video streaming latency and traffic burden on the operator's backhaul. From the MNO's perspective, in-network caching also helps to reduce inter- and intra-MNO traffic and optimize the cost of leasing expensive fiber lines between eNodeBs and EPC [14, 15]. The reduction in outbound traffic from users to content providers (CPs) decreases the traffic load directed to public CDN. This, inherently results in the CP to pay less for CDN services.

Furthermore, by providing multiple source video bit rates for a single video, adaptive bit rate streaming (ABS) increases the wireless network capacity to serve more video requests concurrently [7]. Therefore, combining the advantages of ABS with mobile video caching can maximize the video capacity of the wireless networks, while maintaining or improving the users' video QoE. However, with ABS, each video is divided into multiple chunks and each chunk can be requested at different bit rates. Hence, for an entire video to be served from the cache, one can cache all rate variants. A video could be encoded into more than 40 versions to meet the heterogeneity of user devices and network conditions [16]. This,

in turn, considerably increases backhaul and storage requirements. Moreover, the available transmission rate in a wireless channel is time-varying and hard to predict. Therefore, the selected bitstream transmitted by a content server, distant from the user, may not match the user's transmission characteristics [8].

Alternatively, we can cache only the highest quality of popular videos and use a processing resource to perform video adaptation (rate down-conversion) [7, 8]. The video sequence is encoded at a target bit rate which satisfies a certain quality perception. In this thesis, acceptable video quality is defined as the quality of a stream in presence of only the base quality layer, which is refined by adding enhancement layers. However, it consumes excessive computing and storage resources to encode videos into different bit rates in real-time and store the encoded streams [9, 17].

Additionally, instead of transrating reactively cached video contents, we can perform intelligent caching in the mobile operator network and proactively serve predictable user demands from content caches deployed at mobile gateways. Several approaches have been proposed to analyze intelligent caching strategies for mobile content caching inside MNO's infrastructure [12–14]. In practice, content caches can be installed at multiple levels inside an operator's network [e.g. serving gateway (S-GW), packet data network gateway (P-GW), RAN], leading to an idea of hierarchical in-network caching, which has not be investigated before.

Recently, a new trend of virtualizing mobile network functions into software-based cloud servers has emerged. Thus, with RAN as a Service (RANaaS) paradigm, traditional radio access processing functions are virtualized into a MNO's cloud. Remote antennas [remote radio heads (RRHs)] are connected by high-speed fronthaul fiber networks with the servers running the virtualized baseband units (BBUs) in the operator's cloud center [18]. Likewise, with EPC as a Service (EPCaaS), some EPC network functions are instantiated on virtual machines (VMs), on top of a virtualized platform, running in an operator's cloud

center [19].

The increasing drive towards virtualization of mobile network functions has also motivated the CaaS research, which proposes to implement content caching capacities inside the MNOs' cloud centers [20]. CaaS approach has several advantages over traditional in-network caches and CDNs, e.g. an increase in scalability and flexibility [19–22]. CaaS instances can be adaptively created, migrated, scaled (up or down), shared and released on-demand.

This study aims to propose different methodologies in order to bring the content closer to the end user and maximize the video capacity of mobile networks.

1.1 Thesis Contribution

1.1.1 Key Outcomes

The contributions of this thesis cover different aspects of mobile edge video caching and transrating techniques. The key outcomes of this research in the form of novel solutions and algorithms are summarized below:

- a perceptual quality-aware video adaptation approach for transrating reactively cached videos is proposed. This adaptation scheme provides an empirical mapping between perceptual video quality and source bit rate and encodes a video sequence at a bit rate which satisfies the agreed user requirements of video perceptual quality.

- a resource allocation policy that minimizes power for the target user-perceived video quality is derived such that all users in the network can achieve their target statistical delay bound. The statistical delay QoS requirements are modeled in terms of queue length decaying rate. This can be jointly determined by the effective bandwidth [23] of the arrival traffic and the effective capacity [24] of the wireless channel. The resource allocation problem is

formulated as the minimization of sum power in the downlink, which is subject to perceptual quality guarantees, statistical QoS provisioning as well as the power and resource block (RB) allocation constraints of Orthogonal Frequency Division Multiple Access (OFDMA). A duality-based algorithm is deployed, where dual variables are updated using the efficient ellipsoid method.

- for reactively cached contents, this thesis also proposes a scalable video coding (SVC)-specific active queue management technique which transrates a stream to a lower bit-rate. This scheme drops packets that have minimal negative impact on the user's QoE to satisfy a certain level of QoE for a user. The proposed QoE metric model provides a relationship between packet loss ratio and reduction in QoE and estimates the user QoE of a SVC video, depending on the importance of the video layer that contains the dropped packet. This approach leads to a power-efficient delay-aware resource allocation scheme.

- the problem of storage provisioning for proactive hierarchical in-network video caching is formulated to optimize the trade-off between the cost of transmission bandwidth and the cost of storage. The analysis are focused on SVC-based dynamic adaptive streaming over HTTP (DASH) format. In SVC-based DASH, a video is divided into different layers (base layer and enhancement layers) and when an end user selects the most suitable video representation, different layers are sent over the network as HTTP transactions [25]. Therefore, it is more resource-efficient than traditional H.264/advanced video coding (AVC)-based DASH, which encodes a separate AVC video file for each video quality format [2]. The storage provisioning problem is solved using canonical duality theory (CDT) [26]. More specifically, the formulated binary-integer programming (BIP) problem is

transformed into a canonical dual problem in continuous space, which is a concave maximization problem. Additionally, the conditions under which the solutions of the canonical dual problem and primal problem are identical are provided. The canonical dual problem results in complex non-linear equations which are efficiently solved by applying Invasive Weed Optimization (IWO) algorithm [27].

- this thesis presents the first attempt to formulate a virtual proactive caching optimization framework for AVC-based DASH, which maximizes the return on caching investment. Taking into consideration the popularity and size of video contents, the optimal caching tables which would maximize the ratio of transmission bandwidth cost to storage cost are found. By introducing a quality weighting factor in the optimization problem, the key QoE differentiators in delivering video contents to the end users (e.g. higher throughput, lower latency, smaller start up and buffering times [28, 29]) are taken into account.

1.1.2 List of Publications

The publications[1] related to the main contributions of this thesis are stated as follows. The chapter relevance of different publications is given in TABLE 1.1.

1. **S. E. Ghoreishi**, D. Karamshuk, V. Friderikos, N. Sastry, M. Dohler and A. H. Aghvami, "A Cost-Driven Approach to Caching-as-a-Service in Cloud-Based 5G Mobile Networks" submitted to *IEEE Trans. Mobile Comput.*, March 2016.

2. **S. E. Ghoreishi**, V. Friderikos, D. Karamshuk, N. Sastry and A. H. Aghvami, "Provisioning Cost-Effective Mobile Video Caching", *IEEE Int. Conf. Commun. (ICC)*, May 2016, p. to appear, May 2016.

[1]The numbering does not refer to the Bibliography section of this thesis.

TABLE 1.1: Publications related to individual chapters

Chapter	Journals	Conferences
Chapter 3	(4)	–
Chapter 4	(3)	(5),(6)
Chapter 5	–	(2)
Chapter 6	(1)	–

3. **S. E. Ghoreishi** and A. H. Aghvami, "Power-efficient QoE-Aware Video Adaptation and Resource Allocation for Delay-Constrained Streaming over Downlink OFDMA", *IEEE Commun. Letters*, January 2016.

4. **S. E. Ghoreishi**, A. Aijaz and A. H. Aghvami, "Delay-Constrained Video Transmission: A Power-Efficient Resource Allocation Approach for Guaranteed Perceptual Quality", *Global Commun. Conf. (GLOBECOM)*, pp. 1–7,San Diego, CA, USA, December 2015.

5. **S. E. Ghoreishi**, A. H. Aghvami and M. G. Martini, "Perceptual Quality-Aware Active Queue Management for Video Transmission", *IEEE Int. Symp. Personal, Indoor, Mobile Radio Commun. (PIMRC)*, pp. 1267–1271, Hong Kong, China, September 2015.

6. **S. E. Ghoreishi**, A. H. Aghvami and H. Saki, "Active Queue Management for Congestion Avoidance in Multimedia Streaming," *IEEE European Conf. Netw. Commun. (EuCNC)*, pp. 487–491, Paris, July 2015.

1.2 Thesis Outline

The reminder of the thesis is structured as follows. Chapter 2 provides the preliminaries on the required technical background for understanding the research area addressed. The main contributions of the study are related to four distinct areas: perceptual-quality aware in-network video adaptation; queuing-based QoE-aware in-network video adaptation; provisioning cost-effective video caching and cost-driven CaaS, which are discussed in chapters 3, 4, 5 and 6, respectively. Since each contribution chapter addresses a unique research problem, concluding remarks are presented therein. Based on the overall picture of research conducted in the thesis, the main conclusions and directions for future work are presented in Chapter 7.

Chapter 2

Background Information

2.1 Scalable Video Coding

The rapidly growing number of user devices with different processing capabilities and display features along with the time-varying and potentially unpredictable connection qualities raises the need for a video coding standard that provides multiple video representations with different bit rates, frame rates and frame sizes at relatively low computational cost [30–32].

In video coding, a video sequence is a series of pictures taken at a constant time intervals. Video compression techniques take advantage of the similarities between adjacent pictures to reduce the bit rate. Similar to the the well-known and widespread H.264/AVC standard [33, 34], in SVC, each video sequence is divided into one or more group of pictures (GoP), and each GoP is encoded with three different types of frames: I-frame (intra frame), P-frame (predicted frame) and B-frame (bidirectionally predicted frame). An I-frame contains an entire image and is independently coded without reference to any other frames. A P-frame is coded/decoded using the preceding P-frame or I-frame. Hence, the correct decoding of P-frames depends on the availability of the previous I- or P-frame. On the other hand, a B-frame depends on both preceding and future

frames [35, 36]. Therefore, in a video sequence, I-frames constitute a greater importance in comparison with the P-frames and B-frames.

SVC has been standardized as the scalable extension of H.264/AVC standard by the Joint Video Team (JVT) [37]. In contrast to single-layer bit streams, in a scalable video stream, a lower quality/resolution version of the video can be created just by dropping packets and removing parts of the complete original stream without re-encoding [2, 38, 39]. SVC's capability in reconstructing lower resolution or lower quality signals from bit streams without the need for re-encoding allows for simple solutions in adaptation to network and terminal capabilities [40]. However, adaptation operations, which consist of an adaptation decision and a thinning operation to discard unneeded data are very complex operations [41].

SVC incorporates three scalability modes, providing a full scalability in temporal, quality and spatial dimensions [2, 39, 41, 42]. Temporal scalability allows dividing the video stream frames into a base layer which is coded to provide the basic frame rate and one or multiple enhancement layers. Enhancement layers are coded using temporal prediction corresponding to the base layer or higher enhancement layers. We identify the temporal layers by T_i, starting with T_0 for the base layer and incrementing i by 1 from one temporal layer to the next as shown in Fig. 2.1a. Then, another valid bit stream is formed by removing access units of the temporal layers with $i > k$, where k is a natural number. Decoding both the base layer and all the temporal enhancement layers produces the full temporal resolution video stream.

In spatial scalability, the encoder generates two or more layers with different spatial resolutions from a single video stream. The base layer is encoded independently to deliver the basic spatial resolution and is used by the enhancement layer(s) to provide higher or full spatial resolution of the video [43]. Similar to single-layer coding, SVC deploys motion-compensated prediction and intra prediction in each spatial layer. Moreover, SVC offers inter-layer prediction as shown

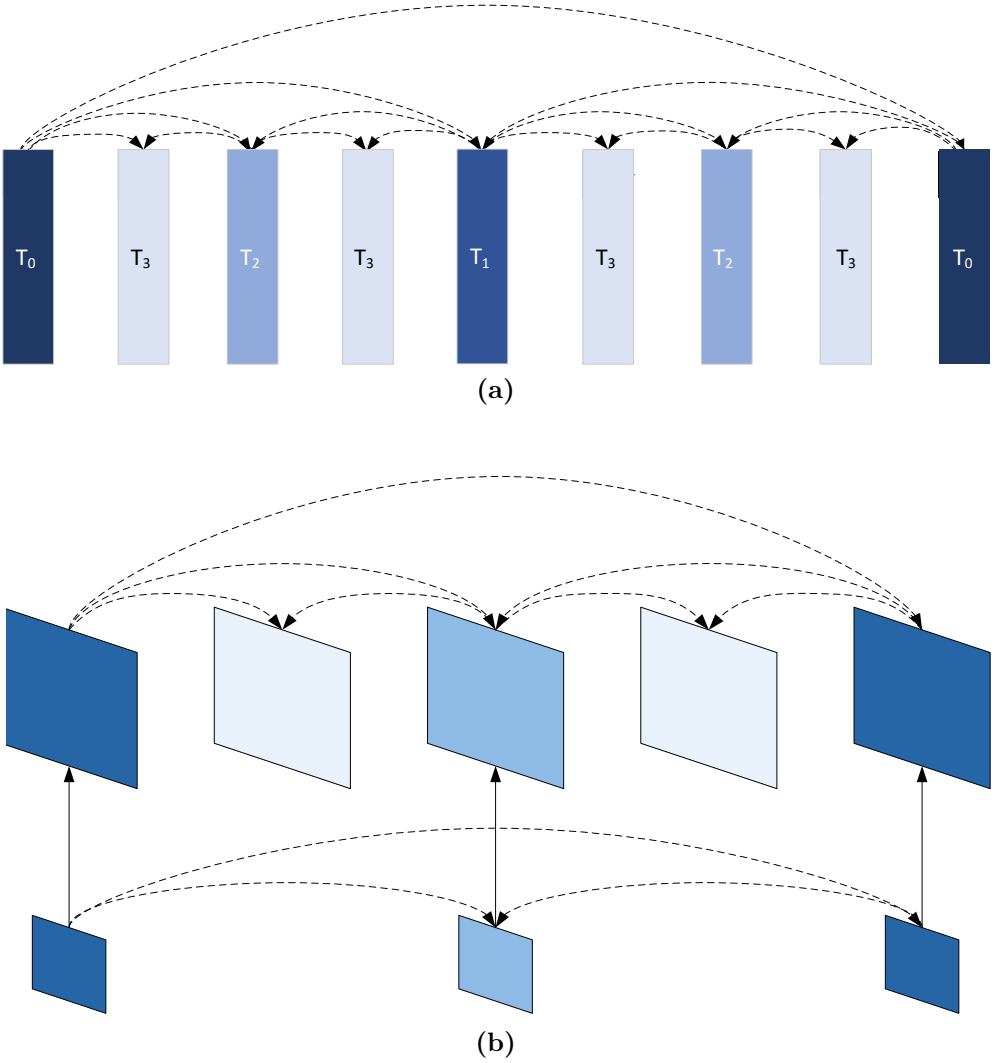


Fig. 2.1: SVC temporal/spatial scalability in a GoP: (a) temporal scalability; (b) spatio-temporal scalability (adapted from [2]).

in Fig. 2.1b, which exploits the statistical dependencies between different spatial layers to enhance the coding efficiency of enhancement layers [2].

In quality scalability, for a single video stream, the encoder generates two or more layers of the same spatial-temporal resolution with different qualities such that the base layer is coded to produce the basic video quality and the enhancement layer(s), when combined with the base layer, reconstruct a higher quality representation of the original video stream and improve the signal to noise ration (SNR) of the base layer.

In addition to a drastic reduction in computational requirements for video

bit rate adaption methods relying on a scalable representation [38, 44], another key advantage of SVC is that rate adaptation can be performed not only at the encoder, but at intermediate network nodes or at the receiver. Therefore, rate adaptation may be applied at the streaming server, intermediate network nodes such as proxy caches or at the receiver [38].

2.2 Dynamic Adaptive Streaming over HTTP

ABS solutions provide an adaptive and dynamic way to stream video. These solutions take into account the available resources such as screen size [45] , CPU load [46] and battery capacity [47], as well as network playout buffer occupancy and bandwidth in order to mitigate stalls and long startup delay caused by fluctuations in available resources [48]. The use of HTTP offers several advantages at both the user- and the server-end. In contrast with traditional video streaming which uses Real-time Transport Protocol (RTP) over UDP, the use of HTTP makes it easier for users to access the content from behind firewalls and network address translator (NAT). Moreover, RTP-based streaming requires the servers to maintain state information for each video session, whereas HTTP video streaming services deploys stateless web servers [49]. Additionally, HTTP streaming moves the control of a streaming session to the client that opens one or more TCP connections with HTTP servers or caches. Therefore, a large number of streaming clients can be provisioned for without increasing the required server resources beyond the standard web usage of HTTP.

DASH [50, 51] is a standard for ABS over HTTP, which is currently used by some of the most popular video streaming services such as YouTube and Netflix [49]. It partitions a video file into one or more segments (typically 2-10 seconds long) [50, 52], which are specified in the media presentation description (MPD) file. It contains information about segment URL addresses, byte ranges,

duration, resolutions, codecs and bit rates. Therefore, using this information, video streaming adaptation is performed by the client who chooses segments based on network conditions, device capabilities, and playout buffer occupancy.

DASH is codec agnostic and is typically integrated with H.264/AVC video codec. Hence, multiple representations of each of the videos are encoded with H.264/AVC at the server and offered side-by-side. However, storing all the representations at the server does not only put a high burden on the storage requirements at the origin server, but might also result in caching inefficiency [53]. With the integration of the more recent codec standard SVC into DASH, all these representations can be embedded in one file [53–57]

2.3 Content Caching

In this section, some of the traditional cache replacement policies are discussed. The main benefits of content caching include reduction in bandwidth usage, user-perceived delays and load on content servers, which makes it attractive to users, network operators and content providers [58, 59]. Due to the limited storage capacity of a content cache, it is infeasible to cache all videos from the original content servers. Therefore, a cache replacement policy (cache algorithm) is defined to manage cache content and choose which content to retain or evict when the cache is full. In general, cache replacement policies aim at making the best use of available resources, including network bandwidth, disk space, server load and processing power [3, 58].

Cache replacement policies can be classified into four main categories of randomized, recency-based, frequency-based, as well as other function-based policies, which take into consideration factors such as time, frequency, size, cost, and latency, and different weighting parameters [60,61]. The most widely used representative cache replacement policies are random replacement (RR), least frequently

used (LFU), least recently used (LRU) and greedy dual size (GD-Size) [61, 62].

RR is a simple randomized cache algorithm that randomly selects a content and evicts it from the cache to make space when necessary [60,63]. As RR requires no state information, it results in both processing power and memory savings. However, policies that use only simple random functions do not achieve good performance in general [3].

With the recency-based algorithm LRU, the assumption is that a content that has been referenced recently are likely to be referenced in the near future [64]. Therefore, the first entry in the cache gets evicted first except when there is a cache hit. In case of a cache hit, the entry is removed from its current position and re-inserted back to the cache, becoming the last entry in the cache. The wide popularity of this rule is primarily due to its satisfactory performance and ease of implementation [3, 65].

The simplistic frequency-based cache replacement policy LFU evicts the least frequently requested content first. It keeps a reference count for each content in the cache and when a content needs to be replaced, the one with the least reference count is evicted. LFU caches sort their entries by their overall popularity rather than their recency. The least popular item is always chosen for eviction. In a situation where there are two contents with the same reference count, it uses LRU to break the tie and removes the least recently used content [66].

On the other hand, the function-based policy GD-Size assigns a priority key to each content arriving at the cache, which takes into account recency, frequency, content size and fetching cost. Cache replacement occurs if an object in the cache has a lower priority key than a newly arrived one [3, 61].

TABLE 2.1 compares the performance of different categories of cache replacement policies in terms of hit ratio (HR), byte hit ratio (BHR) and complexity. HR is the percentage of content requests satisfied by the cache, whereas BHR is based on the number of bytes transferred instead of counting only requests [67].

TABLE 2.1: Categories of cache algorithms and their overall performance (adapted from [3])

Category	Hit Ratio	Byte Hit Ratio	Complexity
Randomized	Worst	Worst	Lowest
Recency-based	Fair	Fair	Fair
Frequency-based	Fair	Fair	Fair
Function-based	Best	Best	Highest

In general, function-based policies perform the best and randomized algorithms perform the worst in terms of HR and BHR. This is mainly explained by the fact that function-based policies take many parameters into account when making replacement decisions, whereas replacement decisions are made randomly for randomized functions. This results in a high complexity for function-based policies and make randomized functions considerably easier to implement [3].

2.4 Resource Allocation in OFDMA Systems

OFDMA is a multiple access version of the popular OFDM and inherits OFDM's resistance to frequency selective fading and inter-symbol interference [68–70]. In OFDMA, in order to deal with frequency-selective fading and support a high data rate, an entire channel is divided into a large number of orthogonal narrow-band sub-channels (subcarriers) [71]. In an OFDMA-based network, different subcarriers can be allocated to different users to provide a flexible multiuser access scheme [72, 73] and exploit multiuser diversity [74]. There is plenty of room to exploit the high degree of flexibility of radio resource management in the context of OFDMA. Due to different channel frequency responses at different frequencies and for different users, data rate adaptation over each subcarrier, adaptive power allocation (APA) and dynamic subcarrier assignment (DSA) result in a considerable improvement in the performance of OFDMA networks. Using data rate

adaptation [75, 76], the transmitter can send higher transmission rates over the subcarriers with better conditions to improve throughput and ensure an acceptable bit-error rate (BER) at each subcarrier. Furthermore, a data transmission rate close to the channel capacity is achievable by optimal power allocation with dynamic subcarrier assignment [77]. However, deep fading on some subcarriers still results in low channel capacity. On the other hand, channel characteristics for different users in multiuser environments are almost independent. Hence, the subcarriers in a deep fade for one user may not be experiencing deep fading for other users [78].

By dynamically assigning subcarriers, the network can benefit from multiuser diversity, which can be exploited by scheduling transmissions when a user has favorable channel conditions [79]. Using this approach, the system capacity increases with the number of users [80]. Exploiting multiuser diversity can also lead to a considerable enhancement in the spectral efficiency. Apart from the spectral efficiency, QoS and fairness are of great importance for resource allocation in wireless networks. Achieving optimality for spectral efficiency, fairness and QoS is usually unfeasible [78]. For instance, throughput-optimal scheduling schemes are unfair to those users faraway from a base station or with bad channel conditions, whereas the absolute fairness may result in low bandwidth efficiency. Thus, it is desirable to achieve an optimal trade-off among efficiency, fairness and QoS in wireless resource allocation (RA).

Chapter 3

Perceptual Quality-Aware In-Network Video Adaptation and Resource Allocation

3.1 Introduction

With the explosive increase of video traffic in mobile networks, it has become necessary to support more simultaneous video streams, while guaranteeing a certain level of quality for individual users. Furthermore, in real-time video transmission, maintaining stringent delay bounds and monitoring perceptual video quality will ensure a good user experience.

By deploying cache proxy servers at the edge of the core network, mobile operators can cache only the best quality of a video and use a processing resource to perform in-network video adaptation [7, 8, 81]. When a user requests to access the video content, the proxy cache transrates it to meet the user's requirement.

Furthermore, deterministic delay bounds are hard to guarantee over wireless networks due to time varying nature of the wireless channel [23]. Therefore, in order to support transmission of real-time applications using OFDMA, statisti-

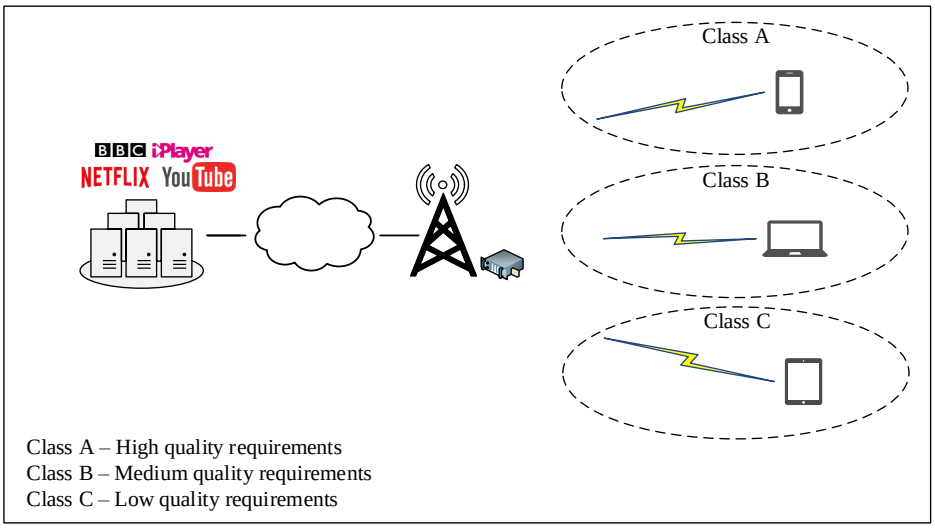


Fig. 3.1: System model

cal delay bound provisioning techniques are considered as a design guideline by defining constraints in terms of the delay-bound violation probability.

This chapter proposes a perceptual quality-aware in-network video adaptation scheme which encodes a video sequence cached at the edge of the network at a target bit rate that satisfies a certain quality of perception. Moreover, it presents a perceptual quality-aware power-efficient resource allocation under statistical delay-bounded QoS guarantees in downlink OFDMA systems. Cross-layer techniques have been adopted in the literature for dynamic resource allocation for downlink OFDMA. However, the common practice is to maximize quality [82], throughput [83, 84] or energy efficiency [84, 85], or to minimize expected distortion [86] or power consumption [87–89].

3.2 Contributions and Outline

The main contributions of this chapter are as follows:

- an empirical mapping between perceptual video quality and source bit rate is provided.
- the statistical delay QoS requirements is modeled in terms of queue length

decaying rate which can be jointly determined by the effective bandwidth [23] of the arrival traffic and the effective capacity [24] of the wireless channel.

- a video specific RA problem is formulated as the minimization of sum power in the downlink. This is subjected to perceptual quality guarantees, statistical QoS as well as the power and RB allocation constraints of OFDMA.

- a duality-based algorithm is used where dual variables are updated using the efficient ellipsoid method.

The rest of this chapter is organized as follows. The system model and problem formulation are presented in Section 3.3. Section 3.4 describes the proposed resource allocation algorithm. Section 3.5 conducts numerical analyses of the model. The conclusion is presented in Section 3.6.

3.3 System Model and Problem Formulation

This chapter focuses on the downlink of 3GPP long term evolution (LTE) networks and considers a single cell, multi-user scenario. The system consists of K mobile users (video streams), which are indexed by the set $\mathcal{K} \triangleq \{1, \ldots, k, \ldots, K\}$. Each video stream k requires a bounded delay of $d_k^{\max}$, a delay violation probability of Γ_k and a source bit rate of $A_k^{\min}$ bits per second which guarantees a target perceptual video quality Q_k. It is assumed that the total number of available RBs are indexed by the set $\mathcal{L} \triangleq \{1, \ldots, l, \ldots, L\}$.

Using the source bit rate adaptation module (Fig. 3.2a), for a user k, the video sequence is encoded at a target bit rate $A_k^{\min}$ which satisfies a certain quality perception Q_k. Afterwards, θ_k and the minimum required data rate, $R_k^{\min}$, are determined in order to guarantee a specific delay QoS requirement given by $(d_k^{\max}, \Gamma_k)$, using the delay-aware data rate adaptation module (Fig. 3.2b). Lastly,

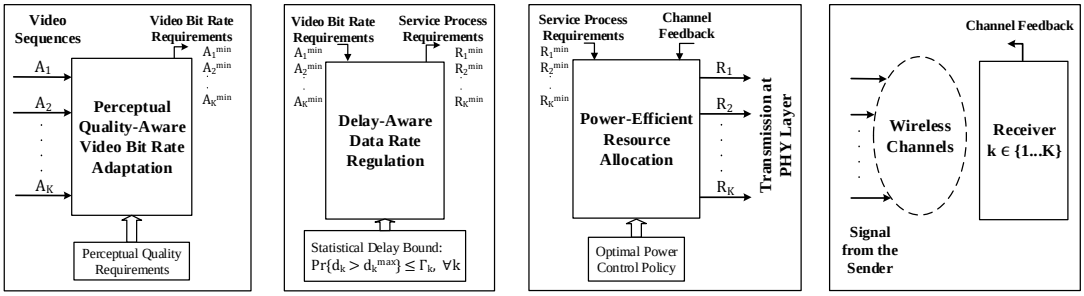


Fig. 3.2: The system modeling framework for video transmission over wireless network: (a) bit rate adaptation module; (b) data rate adaptation module; (c) resource allocation module; (d) receiver.

the resource allocation module (Fig. 3.2c) allocates resources integrating $R_k^{\min}, \theta_k$ and the optimal power control policy $\mu_{k,l}$ presented in Theorem 1.

A summary of commonly used notation is provided in TABLE 3.1

In the following subsections, we concisely explain the modular framework and formulate the resource allocation problem mathematically.

3.3.1 Perceptual Quality-Aware Source Bit Rate Adaptation

PSNR is widely used as a measure of the quality degradation of digitally encoded video. It is calculated as the error between the original and the reconstructed pictures. For a video sequence, PSNR can be derived as $10 \log \left(\frac{255^2}{\frac{1}{N} \sum_{i=1}^{N} \varepsilon^2(i)} \right)$, where $\varepsilon^2(i)$ is the pixel luminance mean-squared error between corresponding frame i in the reference and compressed videos, and N is the number of frames in the degraded video.

Perceptual quality is receiving considerable interest as a method to quantify the multimedia experience of mobile users. [90] relates perceptual quality to PSNR as

$$q_k = \frac{1}{1 + e^{b_1^k (\mathrm{PSNR}_k - b_2^k)}}, \tag{3.1}$$

TABLE 3.1: Commonly Used Notations

Notation	Description
K	Number of video users
θ_k	Statistical delay exponent of user k
L	Number of RBs
Q_k	Target perceptual quality
$d_k^{\max}$	Delay bound of user k
B	RB bandwidth
Γ_k	Delay violation probability of user k
$\mathcal{P}_{k,l}$	Transmit power of user k over l^{th} RB
$R_k^{\min}$	Minimum data rate of user k
$\mathcal{B}_E(\theta_k)$	User k's effective bandwidth
$P^{\max}$	Transmit power upper bound
$\mathcal{C}_E(\theta_k)$	User k's effective capacity
$A_k^{\min}$	Minimum video source rate of user k
$\mu_{k,l}$	Optimal power control policy for the k^{th} user over RB l
$\gamma_{k,l}$	SNR of user k over the l^{th} RB
$x_{k,l}$	RB indicator vector

where b_1^k and b_2^k are parameters that depend on the video characteristics. In (3.1), $q_k = 0$ indicates the best quality and $q_k = 1$ indicates the worst quality. A perceptual quality metric derived in [91], based on the metric in [90], is expressed as

$$Q_k = q_{\max}^k \left(1 - \frac{1}{1 + e^{b_1^k(\mathrm{PSNR}_k - b_2^k)}}\right) \frac{1 - e^{-b_3^{(k)} \frac{f^{(k)}}{f_{\max}^{(k)}}}}{1 - e^{-b_3^{(k)}}}, \tag{3.2}$$

where b_1^k, b_2^k and $b_3^{(k)}$ are parameters that depend on the video characteristics, $q_{\max}^k$ is a constant corresponding to maximum quality, $f^{(k)}$, is the frame rate at which the video is displayed and $f_{\max}^{(k)}$ is the maximum frame rate.

Source video sequences are encoded and multiple bit rate versions of each video content (different levels of quality) are produced. The error concealment method proposed in [92] is deployed in order to make (3.2) suitable to assess transmission over wireless systems and maintain the same frame rate after error concealment ($f^{(k)} = f_{\max}^{(k)}$). Therefore, the target perceptual video quality of the k^{th} stream, (3.2), can be simplified to

$$Q_k = q_{\max}^k \left(1 - \frac{1}{1 + e^{b_1^k(\mathrm{PSNR}_k - b_2^k)}}\right), \tag{3.3}$$

where $q_{\max}^k = 100$, thus displaying perceptual quality on a scale from 0 to 100.

For each user k, the required minimum video source rate that satisfies the user's perceptual quality requirements is determined. Therefore, having encoded the source streams at multiple bit rates, the PSNR variation with the bit rate is measured. For each encoded bit rate, the corresponding user-perceived quality is calculated using (3.3). Repeating over all sequences, an empirical mapping between perceptual video quality and source bit rate is provided. Therefore, for each user k, the minimum bit rate $A_k^{\min}$ that satisfies the user perceptual quality requirements is found.

3.3.2 Optimal Data Rate Adaptation for Statistical Delay QoS Guarantees

Statistical QoS guarantees have been extensively investigated in literature in the context of effective bandwidth $\mathcal{B}_E$ and effective capacity $\mathcal{C}_E$ functions [23, 24]. The effective bandwidth is defined as the minimum constant service rate required by a given arrival process for which a statistical QoS requirement specified by θ_k is fulfilled. θ_k characterizes the queue length decaying rate. Inspired by the effective bandwidth, [23] proposed effective capacity. The effective capacity is defined as the maximum constant arrival rate that a given service process can support in order to guarantee statistical delay-QoS requirements specified by θ_k. Specifically, for a dynamic queuing system, under sufficient conditions, the queue length process, $\mathcal{Q}(t)$, converges in distribution to a random variable $\mathcal{Q}(\infty)$ such that [93]

$$-\lim_{z_t \to 0} \frac{\ln(Pr\{\mathcal{Q}(\infty) > z_t\})}{z_t} = \theta_k. \tag{3.4}$$

The above equation states that the probability of the queue length exceeding a certain threshold z_t decays exponentially fast as z_t increases and the parameter θ_k determines the decaying rate.

Considering a discrete-time arrival process $\{A[i], i = 1, 2, \cdots\}$ and the time-accumulated arrival process $S_B[t] \triangleq \sum_{i=1}^{t} A[i]$, effective bandwidth can be expressed as

$$\mathcal{B}_E(\theta_k) = \lim_{t \to \infty} \frac{1}{t\theta_k} \log\left(\mathbb{E}\left\{e^{\theta_k S_B[t]}\right\}\right), \tag{3.5}$$

where $\mathbb{E}\{\cdot\}$ denotes the expectation. Moreover, for effective bandwidth, the prob-

ability of delay-bound violation can be approximated as [24]

$$Pr\{d_k > d_k^{max}\} \approx e^{-\theta_k \mathcal{B}_E(\theta_k) d_k^{max}} \leq \Gamma_k, \tag{3.6}$$

where $d_k^{\max}$ and Γ_k are the delay-bound and delay violation probability thresholds for a user k. Likewise, given a discrete-time, stationary and ergodic stochastic service process $\{R[i], i = 1, 2, \cdots\}$ and the time-accumulated service process $S_C[t] \triangleq \sum_{i=1}^{t} R[i]$, effective capacity is given by [23, 24]

$$\mathcal{C}_E(\theta_k) = -\lim_{t\to\infty} \frac{1}{t\theta_k} \log\left(\mathbb{E}\left\{e^{-\theta_k S_C[t]}\right\}\right). \tag{3.7}$$

The time-frame index [i] is dropped for the corresponding variables to simplify notations. Since the service rate $R_k^{\min}$ is a stationary and ergodic process that is uncorrelated across different time frames, the effective capacity formulation simplifies to [94]

$$\mathcal{C}_E(\theta_k) = -\frac{1}{\theta_k} \log\left(\mathbb{E}\left\{e^{-\theta_k R_k^{\min}}\right\}\right). \tag{3.8}$$

The statistical delay guarantees is modeled in terms of QoS exponent, effective bandwidth/capacity, and delay-bound violation probability as in [94]. For a given arrival process $A_k^{\min}$ determined in section 3.3.1, we get the corresponding effective bandwidth using (3.5). (3.6) is then applied to calculate the solution QoS exponent θ_k^* that guarantees a specific delay QoS requirement given by $(d_k^{\max}, \Gamma_k)$. Having found θ_k^* and $\mathcal{B}_E(\theta_k^*)$, the corresponding data rate, R_k^{min}, is designed such that $\mathcal{C}_E(\theta_k^*) \geq \mathcal{B}_E(\theta_k^*)$ is satisfied

$$-\frac{1}{\theta_k^*} \log\left(\mathbb{E}\left\{e^{-\theta_k^* R_k^{\min}}\right\}\right) \geq \mathcal{B}_E\left(\theta_k^*\right). \tag{3.9}$$

Rate Requirements

The SNR for the k^{th} user over the l^{th} RB is given by $\gamma_{k,l} = \frac{P_{k,l}|h_{k,l}|^2}{\sigma^2}$, where $P_{k,l}$ is the transmission power of the k^{th} user over the l^{th} RB, $h_{k,l}$ is the channel fading coefficient, and σ^2 denotes the power of additive white Gaussian noise (AWGN). The item $\frac{|h_{k,l}|^2}{\sigma^2}$ is called CNR, which fully reflects the quality of each wireless channel. Perfect channel state information (CSI) is assumed at both base station (BS) and each user, which enables BS to dynamically allocate power and rate on each tone according to channel conditions.

Using Shannon's capacity formula, the upper bound on the achievable service rate for the k^{th} user over the l^{th} RB, denoted by $R_{k,l}$ can be expressed as

$$R_{k,l} = B \log_2 (1 + \frac{\mu_{k,l} P_{k,l} |h_{k,l}|^2}{\sigma^2}), \tag{3.10}$$

where B is the bandwidth of each RB and $\mu_{k,l}(\theta_k, \gamma_{k,l})$ denotes the optimal power control policy to be discussed later. Applying the power adaptation, the instantaneous transmit power becomes

$$\mathcal{P}_{k,l} = \mu_{k,l}(\theta_k, \gamma_{k,l}) P_{k,l} \qquad \forall l \in \mathcal{L}, \forall k \in \mathcal{K}. \tag{3.11}$$

Power Control Policy

The power control policy, denoted by $\mu_{k,l}(\theta_k, \gamma_{k,l})$, gives the relationship between R^k , θ_k and allocated power. Conventionally, the power control policy is expressed as a function of SNR only. However in this case, it is a function of both SNR and QoS exponent.

Theorem 1. *The optimal power control policy [83] for the k^{th} user over the l^{th}*

RB, denoted by $\mu_{k,l}(\theta_k, \gamma_{k,l})$ can be expressed as

$$\mu_{k,l}(\theta_k, \gamma_{k,l}) = \frac{1}{\gamma_{k,l}} \left[\left(\frac{\gamma_{0_{k,l}}}{\gamma_{k,l}} \right)^{\frac{1}{q_k - 1}} - 1 \right]^+, \tag{3.12}$$

where $[x]^+ = \max(0, x)$, $q_k = -\frac{\theta_k B}{\ln 2}$ is defined as the normalized QoS exponent and $\gamma_{0_{k,l}}$ is the cutoff SNR.

Proof. The proof directly extends from [83]. ■

3.3.3 Problem Formulation

The resource allocation problem is mathematically formulated as

$$\min_{P_{k,l}, x_{k,l}} \; P_s = \sum_{k=1}^{K} \sum_{l=1}^{L} \mathcal{P}_{k,l} x_{k,l} \tag{3.13}$$

subject to:

$$\sum_{l=1}^{L} B \log_2(1 + \frac{\mathcal{P}_{k,l} |h_{k,l}|^2}{\sigma^2}) x_{k,l} \geq R_k^{\min} \qquad \forall k \in \mathcal{K} \tag{3.13a}$$

$$\sum_{k=1}^{K} \sum_{l=1}^{L} \mathcal{P}_{k,l} x_{k,l} \leq P^{\max} \tag{3.13b}$$

$$\sum_{k=1}^{K} x_{k,l} \leq 1 \qquad \forall l \in \mathcal{L} \tag{3.13c}$$

$$x_{k,l} \in \{0, 1\}, \; \mathcal{P}_{k,l} \geq 0 \qquad \forall k \in \mathcal{K}, \forall l \in \mathcal{L}. \tag{3.13d}$$

The objective of the optimization problem in (3.13) is power and RB allocation for different users in order to minimize the cumulative transmit power, P_s, in the downlink. It is subject to different constraints of OFDMA along with satisfying the statistical delay-bound, maximum transmission power and data rate requirements. R_k^{min} in (3.13a) is the minimum required data rate for the delay constrained video services of receiver k, specified in Section 3.3.2. The value of $P^{\max}$ in (3.13b) puts an upper limit on the power radiated by the transmitter.

(3.13c) and (3.13d) indicate that each RB can be allocated to one receiver exclusively. Binary variables $x_{k,l} \in \{0,1\}$ is used to represent the RB assignment in multi-user systems, where $x_{k,l} = 1$ indicates RB l is used to serve user k and $x_{k,l} = 0$ otherwise.

3.4 Duality-Based Resource Allocation

In this section, some desirable properties of the optimal solution are derived, and (3.13) is solved using dual decomposition. The Lagrangian of problem (3.13) is given by

$$\begin{aligned}\boldsymbol{\mathcal{L}}(\mathbf{X},\mathbf{P},\lambda,\boldsymbol{\nu}) =& \sum_{k=1}^{K}\sum_{l=1}^{L}\mathcal{P}_{k,l}x_{k,l} + \lambda\left(\sum_{k=1}^{K}\sum_{l=1}^{L}\mathcal{P}_{k,l}x_{k,l} - P^{\max}\right) \\ &+ \sum_{k=1}^{K}\nu_k\left(R_k^{min} - \sum_{l=1}^{L}R_{k,l}x_{k,l}\right) \\ =& \sum_{l=1}^{L}\left[\sum_{k=1}^{K}(1+\lambda)\,\mathcal{P}_{k,l}x_{k,l} - \sum_{k,l}^{K}\nu_k R_{k,l}x_{k,l}\right] \\ &+ \sum_{k=1}^{K}\nu_k R_k^{min} - \lambda P^{\max}, \end{aligned} \tag{3.14}$$

where $\mathbf{X}$ and $\mathbf{P}$ are both $K \times L$ matrices with elements $x_{k,l}$ and $P_{k,l}$, respectively. λ is the dual variable for the power constraint and $\boldsymbol{\nu} = [\nu_1, \ldots, \nu_k, \ldots, \nu_K]$ is the dual vector for the data rate constraint. The Lagrangian dual function $g(\lambda, \boldsymbol{\nu})$ is defined as

$$g(\lambda,\boldsymbol{\nu}) = \begin{cases} \min\limits_{\boldsymbol{X},\boldsymbol{P}} \ \boldsymbol{\mathcal{L}}(\mathbf{X},\mathbf{P},\lambda,\boldsymbol{\nu}) \\ \text{subject to:} \\ \sum\limits_{k=1}^{K} x_{k,l} \le 1 \qquad\qquad \forall l \in \mathcal{L} \\ x_{k,l} \in \{0,1\},\ \mathcal{P}_{k,l} \ge 0 \quad \forall k \in \mathcal{K}, \forall l \in \mathcal{L}, \end{cases} \tag{3.15}$$

and the dual problem is

$$G = \max_{\lambda \geq 0, \boldsymbol{\nu} \succeq 0} g(\lambda, \boldsymbol{\nu}). \tag{3.16}$$

In general, there is a non-zero duality gap in presence of integer constraints. However, when time-sharing condition is satisfied, we have an asymptotically zero duality gap as L goes to infinity, and for practical systems with finite L, the duality gap is still nearly zero [95]. Via Lagrangian relaxation (3.14), we have removed the coupling among RBs. Thus, $g(\lambda, \boldsymbol{\nu})$ is decomposed into L sub-problems which can be independently solved at each RB, given $(\lambda, \boldsymbol{\nu})$. The sub-problem at RB l is

$$\min_{\boldsymbol{X}_l, \boldsymbol{P}_l} \boldsymbol{\mathcal{L}}_l(\mathbf{X}_l, \mathbf{P}_l) = \sum_{k=1}^{K} \mathcal{P}_{k,l} x_{k,l} + \sum_{k=1}^{K} \lambda \mathcal{P}_{k,l} x_{k,l} - \sum_{k,l}^{K} \nu_k R_{k,l} x_{k,l} \tag{3.17}$$

subject to:

$$\sum_{k=1}^{K} x_{k,l} \leq 1,\ x_{k,l} \in \{0,1\},\ \mathcal{P}_{k,l} \geq 0 \qquad \forall l \in \mathcal{L}, \tag{3.17a}$$

where $\mathbf{X}_l$ and $\boldsymbol{\mathcal{P}}_l$ are vectors of $x_{k,l}$ and $\mathcal{P}_{k,l}$ at RB l. By visiting the constraints in (3.17), we note that $\mathbf{X}_l$ is an all-zero vector except for one binary non-zero entry. Hence, the optimal value of

$$\mathcal{F}_{k,l} = \begin{cases} \min\limits_{\mathbf{P}_l} \ (1+\lambda)\mathcal{P}_{k,l} - \nu_k R_{k,l} \\ \text{subject to:} \\ \mathcal{P}_{k,l} \geq 0 \qquad\qquad\qquad \forall\ k \in \mathcal{K}, \end{cases} \tag{3.18}$$

is fist calculated at each l and then optimality is found for sub-problem l within the vector $\boldsymbol{\mathcal{F}}_l = [\mathcal{F}_{1,l}, \mathcal{F}_{2,l}, \cdots, \mathcal{F}_{K,l}]$. Therefore, the scheduling vector $\boldsymbol{X}_l$ for RB

l is derived as

$$x_{k,l} = \begin{cases} 1 & k = k^* = \arg\min_k \mathfrak{F}_l, \mathcal{P}^*_{k,l} \neq 0 \\ 0 & \text{otherwise.} \end{cases} \tag{3.19}$$

Substituting (3.10), (3.11) and (3.12) into (3.18), we have

$$\mathcal{F}_{k,l} = \begin{cases} \min\limits_{\mathbf{P}_l} \ (1+\lambda)\,\mu_{k,l} P_{k,l} \\ -\nu_k B \log_2\left(1 + \frac{\mu_{k,l} P_{k,l} |h_{k,l}|^2}{\sigma^2}\right) \\ \text{subject to:} \\ \mathcal{P}_{k,l} \geq 0 \qquad\qquad \forall\ k \in \mathcal{K}. \end{cases} \tag{3.20}$$

By taking derivative with respect to $P_{k,l}$, the optimal $P_{k,l}$ allocation on RB l is obtained as

$$P^*_{k,l} = \begin{cases} \gamma_{0_{k,l}} \left(\frac{\sigma^2}{|h_{k,l}|^2}\right) \left(\frac{\nu_k}{(1+\lambda)} \frac{B}{\ln 2} \frac{|h_{k,l}|^2}{\sigma^2} + 1\right)^{1-q_k} & k = k^* \\ 0 & \text{otherwise.} \end{cases} \tag{3.21}$$

By updating the dual vector $(\lambda, \boldsymbol{\nu})$ at each iteration, the Ellipsoid Method [96] can efficiently solve dual problem (3.16) and achieve dual optimality $(\lambda^*, \boldsymbol{\nu}^*)$. The subgradient is required by ellipsoid method at each iteration. The subgradient at the n^{th} iteration is derived in the following proposition.

Proposition 1. *For the optimization problem* (3.13) *with dual defined in* (3.16)*, a subgradient for* $g(\lambda, \boldsymbol{\nu}))$ *is*

$$\begin{aligned} d(\lambda_k(n)) =& \sum_{k=1}^{K} \sum_{l=1}^{L} \mathcal{P}^*_{k,l}(n) x_{k,l}(n) - P^{\max} \\ d(\nu_k(n)) =& R_k^{min} - \sum_{l=1}^{L} R^*_{k,l}(n) x_{k,l}(n), \end{aligned} \tag{3.22}$$

where $\mathcal{P}^*_{k,l}(n) = \mu_{k,l}(\theta_k, \gamma^*_{k,l})P^*_{k,l}$ *and* $R^*_{k,l}(n) = B\log_2(1 + \frac{\mu_{k,l}(\theta_k,\gamma^*_{k,l})P^*_{k,l}|h_{k,l}|^2}{\sigma^2})$. $P^*_{k,l}$ *minimizes* (3.15) *at* λ *and* $\boldsymbol{\nu}$.

Proof. By definition of $g(\lambda, \boldsymbol{\nu})$ in (3.15)

$$\begin{aligned} g(\lambda', \boldsymbol{\nu}') \leq & \sum_{k=1}^{K}\sum_{l=1}^{L}\mathcal{P}^*_{k,l}x_{k,l} + \lambda'\left(\sum_{k=1}^{K}\sum_{l=1}^{L}\mathcal{P}^*_{k,l}x_{k,l} - P^{\max}\right) \\ & + \sum_{k=1}^{K}\nu'_k\left(R^{min}_k - \sum_{l=1}^{L}R^*_k x_{k,l}\right) \\ & = g(\lambda, \boldsymbol{\nu}) + (\lambda' - \lambda)\left(\sum_{k=1}^{K}\sum_{l=1}^{L}\mathcal{P}^*_{k,l}x_{k,l} - P^{\max}\right) \\ & + \sum_{k=1}^{K}(\nu'_k - \nu_k)\left(R^{min}_k - \sum_{l=1}^{L}R^*_k x_{k,l}\right). \end{aligned} \tag{3.23}$$

Thus, proposition 1 is proven using subgradient definition. ■

Lemma 1. *The optimal dual variables* $(\lambda^*, \boldsymbol{\nu}^*)$ *must satisfy*

$$0 \leq \nu^*_k \leq \nu^{\max}_k = \frac{\ln 2}{B}\mu_\alpha P_{k,l}\,(1 + \lambda^{\max}) \qquad \forall k \in \mathcal{K}, \tag{3.24}$$

$$0 \leq \lambda^* \leq \lambda^{\max} = \frac{B}{\ln 2}\frac{\nu^*}{\mu_\beta P_{k,l}}, \tag{3.25}$$

where μ_α *and* μ_β *are the total channel inversion [75], [97] and water-filling [75], [98] power-control policies, respectively.*

Proof. The dual variables $(\lambda^*, \boldsymbol{\nu}^*)$ must satisfy the Karush-Kuhn-Tucker (KKT) conditions in order to be optimal. Taking the partial derivative of (3.17) at RB l with respect to $P_{k,l}$, we obtain (3.26) and (3.27).

$$\begin{aligned}\nu_k^* =& \frac{\ln 2}{B}\left(\frac{|h_{k,l}|^2}{\sigma^2}\right)^{\frac{q_k}{1-q_k}}\left(\frac{P_{k,l}}{\gamma_{0_{k,l}}}\right)^{\frac{1}{1-q_k}}(1+\lambda^*) - \frac{\ln 2}{B}\frac{\sigma^2}{|h_{k,l}|^2}(1+\lambda^*) \\ =& \frac{\ln 2}{B}\frac{1}{\gamma_{k,l}}\left[\left(\frac{\gamma_{k,l}}{\gamma_{0_{k,l}}}\right)^{\frac{1}{1-q_k}} - 1\right]P_{k,l}(1+\lambda^*) = \frac{\ln 2}{B}\mu_{k,l}P_{k,l}(1+\lambda^*),\end{aligned} \tag{3.26}$$

$$\lambda^* = \frac{B}{\ln 2}\frac{\nu_k^*}{\frac{1}{\gamma_{k,l}}\left[\left(\frac{\gamma_{k,l}}{\gamma_{0_{k,l}}}\right)^{\frac{1}{1-q}} - 1\right]P_{k,l}} = \frac{B}{\ln 2}\frac{\nu_k^*}{\mu_{k,l}P_{k,l}}. \tag{3.27}$$

$\mu_{k,l}$ is upper-bounded by the channel inversion scheme denoted by μ_α and lower-bounded by the water-filling power adoption denoted by μ_β [83]. Therefore, the upper bound $\nu_k^{\max}$ is obtained by letting $\mu_{k,l} = \mu_\alpha$ and $P_{k,l} = P^{\max}$ in (3.26). Likewise, the upper bound $\lambda^{\max}$ is derived by substituting $\nu_k^{\max}$ and $\mu_{k,l} = \mu_\beta$ into (3.27). ■

Using Lemma 1, one may choose an initial ellipsoid $\mathbf{A}(0)$ with a center $\mathbf{z}(0)$ in which the optimal $(\lambda^*, \boldsymbol{\nu}^*)$ reside. The details, e.g. the update algorithm and stopping criterion can be found in [96].

A summary of the proposed algorithm is provided in Algorithm 1.

3.5 Numerical and Simulation Results

In the following simulations, the downlink of a single-cell OFDMA system is considered. The system bandwidth is 10 MHz. Therefore, 50 usable RBs are available per transmission time interval (TTI). The channel model accounts for small scale Rayleigh fading, large scale path loss [99], and shadowing (log-normally distributed). 8 uniformly distributed users are considered in the coverage area with a minimum distance of 50 m from the eNodeB.

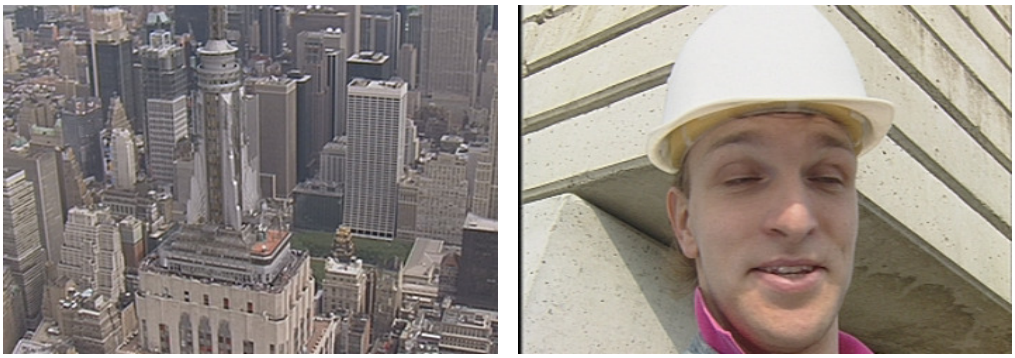
joint scalable video model (JSVM) 9.19.15 [100] is used to encode/decode

Algorithm 1: Power-efficient resource allocation

initialize $(\lambda(0), \boldsymbol{\nu}(0))$ and the initial ellipsoid, $\boldsymbol{A}(0)$; **repeat**
  initialize $P_{k,l}$;
  for $l = 1$ *to* L **do**
    for $k = 1$ *to* K **do**
      Obtain the optimal $P_{k,l}$ through (3.21);
      Calculate vector $\mathcal{F}_{k,l}$ in (3.18) with optimal $P_{k,l}$;
      Get the optimal assignment for RB l by (3.19);
    end
  end
  Update $(\lambda, \boldsymbol{\nu})$ and $\boldsymbol{A}$ via the ellipsoid method with the subgradients in (3.22);
until *$(\lambda, \boldsymbol{\nu})$ convergence*;

the video streams. The common intermediate format (CIF) (352×288) video sequences "city" and "foreman" are used in the simulations. The parameters (b_1^k, b_2^k) are set to (0.34, 29.09) for "foreman" and (0.34, 26.3) for "city" [91]. Multiple bit rates of the videos are generated in order to produce different levels of PSNRs (see Fig. 3.4a). Using (3.3), Fig. 3.4b presents an empirical mapping between PSNR and perceptual quality.

In order to analyze the proposed approach, three scenarios are considered, in



(a) **(b)**

Fig. 3.3: SVC sequences investigated: a) City; b) Foreman.

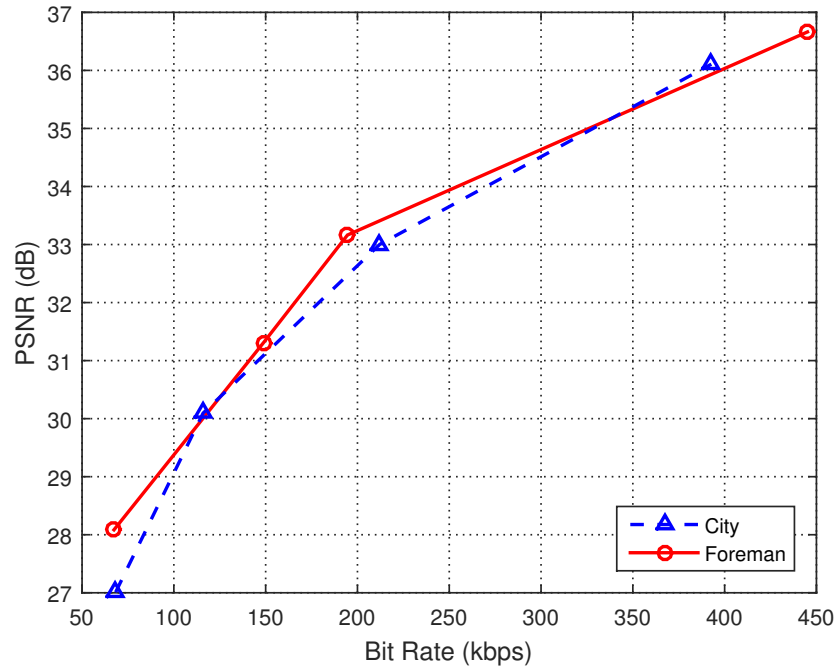
TABLE 3.2: Simulation Configuration Parameters

Parameter	**Value**
Cell radius	1 km
Path loss	$128.1 + 37.6 \log_{10}(r)$ dB, r in km
Standard deviation of shadowing	8 dB (90% cell edge coverage)
Quality requirements for different scenarios	
Scenario 1	Foreman sequence, $d_1^{max} = 150$ ms, $\Gamma_1 = 10^{-2}$, $Q_1 \approx 70$
Scenario 2	Foreman sequence, $d_2^{max} = 100$ ms, $\Gamma_2 = 10^{-3}$, $Q_2 \approx 80$
Scenario 3	City sequence, $d_3^{max} = 70$ ms, $\Gamma_3 = 10^{-4}$, $Q_3 \approx 90$

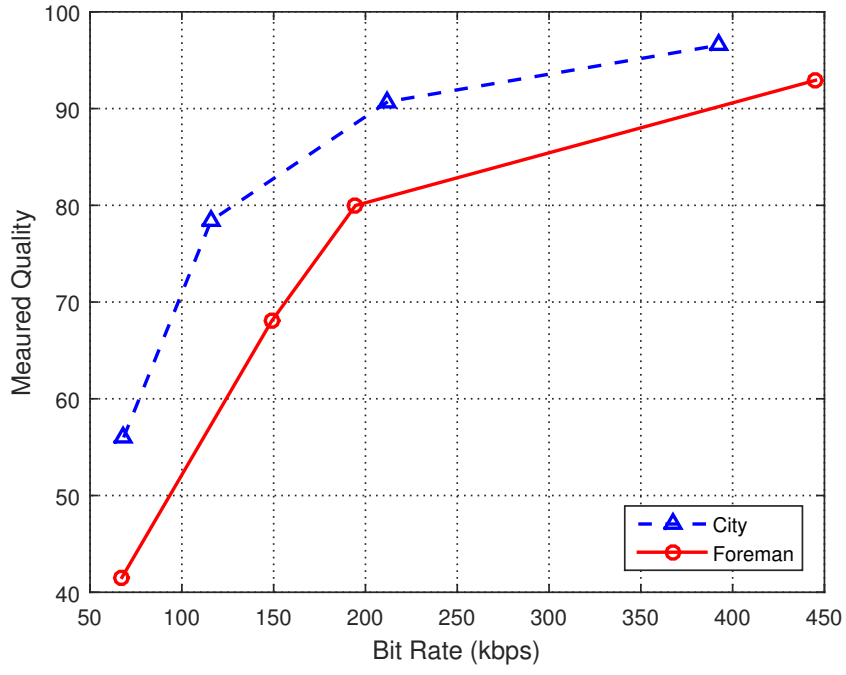
each of which, users have different quality and delay requirements as shown in TABLE 3.2. *Scenario 1* has the highest and *Scenario 3* has the lowest quality requirements. The performance of the proposed algorithm is evaluated on these scenarios.

The proposed algorithm is compared with WSPmin scheme in [87] and VAWS method in [13]. WSPmin minimizes the total transmission power with a minimum rate constraint on each user. In VAWS, subcarriers are assigned to satisfy minimum rate constraint with the assumption of equal power allocation per subcarrier. It then refines the initial uniform power allocation given the subcarrier assignment in the last stage to ensure that minimum rate requirements are met. It lastly repeats the previous phases to refine power allocation. Nevertheless, WSPmin and VAWS do not provide statistical delay QoS guarantees. The data rate requirements for the users served by WSPmin and VAWS are randomly varying from 100 kbps – 400 kbps as multiples of 50 kbps.

Fig. 3.5 plots the heatmaps of θ_k for a user k in *Scenarios 1-3* for different delay QoS requirements $(d_k^{\max}, \Gamma_k)$. The θ_k for the target delay bounds and target delay bound violation probabilities in *Scenarios 1-3* are highlighted on Figs. 3.5a-



(a)



(b)

Fig. 3.4: Perceptual quality-rate mapping: (a) PSNR vs. bit rate. (b) quality vs. bit rate.

3.5c.

Fig. 3.6 illustrates the sum power for different scenarios and resource allocation schemes under different average cell border CNRs. As in [85], the noise variance σ^2 is set to ensure average cell border CNR ρ_0. We see that in terms of power efficiency, the proposed method performs considerably better than WSPmin and VAWS . For instance, in *Scenario 1*, with an average cell border CNR of 4 dB, the proposed approach can improve power efficiency by 52.9% compared with WSPmin and 72.2% compared with VAWS.

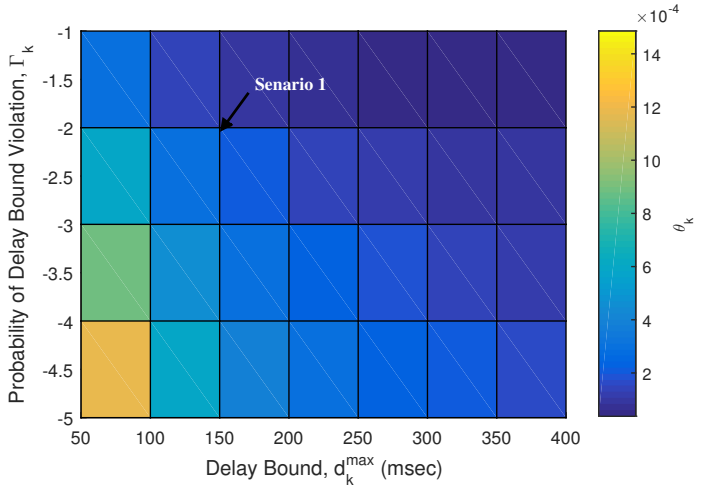
It is also noted that the power efficiency decreases as the video quality requirements increases. This is due to the fact that more power is allocated per RB in order to provide a more stringent delay QoS guarantee and satisfy the higher perceptual quality requirements. For instance, in *Scenario 3* which has higher quality requirements, with the same average cell border CNR of 4 dB, power efficiency is improved by 30.2% and 55.3% compared with WSPmin and VAWS, respectively.

3.5.1 Complexity Analysis

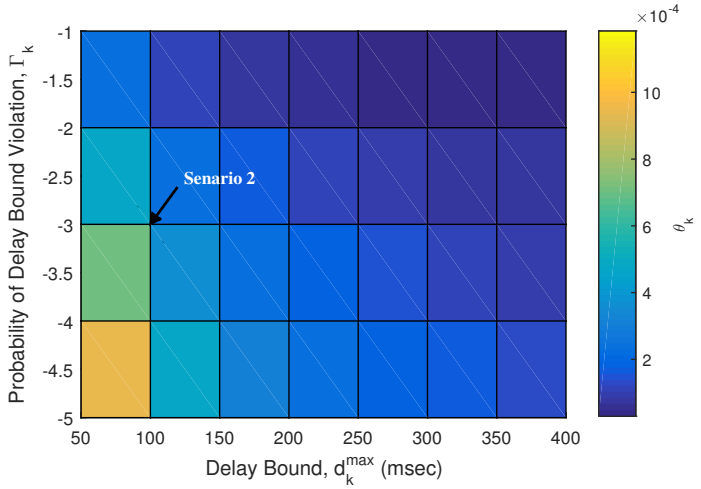
The complexity to solve all sub-problems in (3.17) is $\mathcal{O}(KL)$. Therefore, the complexity of ellipsoid method with $(K+1)$ dual variables is $\mathcal{O}(KL(K+1)^2)$ [87]. The overall complexity can be estimated by $\mathcal{O}(KL(K+1)^2 \log_2(\frac{1}{\epsilon}))$ where ϵ is the required accuracy (polynomial complexity).

3.6 Conclusion

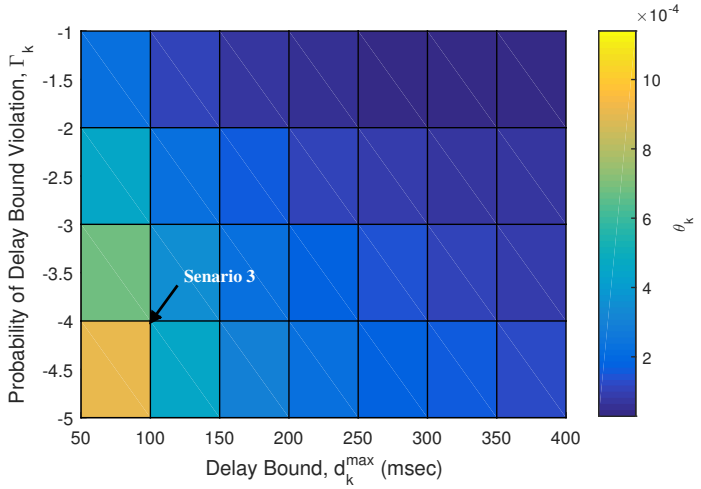
This chapter has proposed a perceptual in-network quality-aware video adaptation scheme which encodes a video sequence cached at the edge of the network at a target bit rate that satisfies a certain quality of perception. It has also investigated power efficient resource allocation for the downlink of LTE networks



(a)



(b)



(c)

Fig. 3.5: Probability of delay violation of user k: (a) *Scenario 1*; (b) *Scenario 2*; (c) *Scenario 3* (y-axes in logarithmic scale).

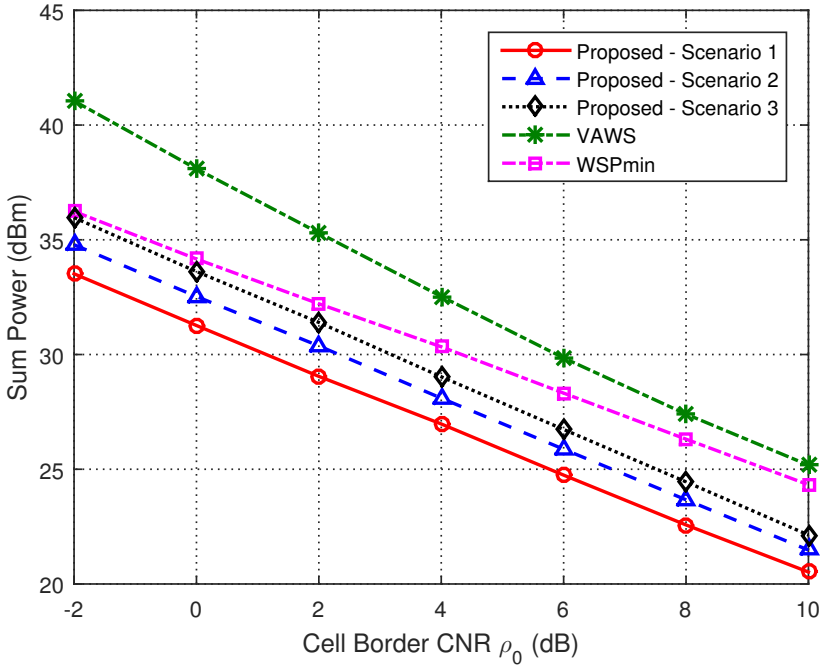


Fig. 3.6: Sum power versus border CNR, ρ_0.

under user-perceived quality and statistical delay QoS constraints. The resource allocation problem has been solved using a duality-based approach. Numerical and simulation results have shown that the proposed resource allocation algorithm not only outperforms classical algorithms in terms of power efficiency but also satisfies the QoS requirements of different users for the target perceptual qualities.

Next chapter proposes a queuing-based QoE-aware in-network video adaptation and resource allocation approach. In the adaptation scheme, packets are dropped selectively from video streams to produce lower bit-rate versions under QoE and delay constraints. Additionally, the resource allocation technique minimizes the transmit power by considering the delay requirements of each stream identified in the video adaptation phase.

Chapter 4

Queuing-Based QoE-Aware In-Network Video Adaptation and Resource Allocation

4.1 Introduction

In Chapter 3, a perceptual quality-aware video adaptation scheme, in addition to a power efficient delay-aware resource allocation apporach were proposed. However, it consumes tremendous computing to encode videos into different bit rates in real-time [9].

In this chapter, as shown in Fig. 4.1, the RAN is enhanced with a queuing-based SVC video adaptation/ RA module. By eliminating the need for downloading and caching multiple bit rate versions of a video, this reduces the cache storage requirements and the load on the RAN backhaul. The module deploys a delay-constrained SVC-specific active queue management technique, which adapts a stream to a lower bit rate and leads to a power-efficient RA scheme. It drops packets that have minimal negative impact on the user's QoE to satisfy a certain level of QoE for a user. This, in turn, reduces network load and delay, and

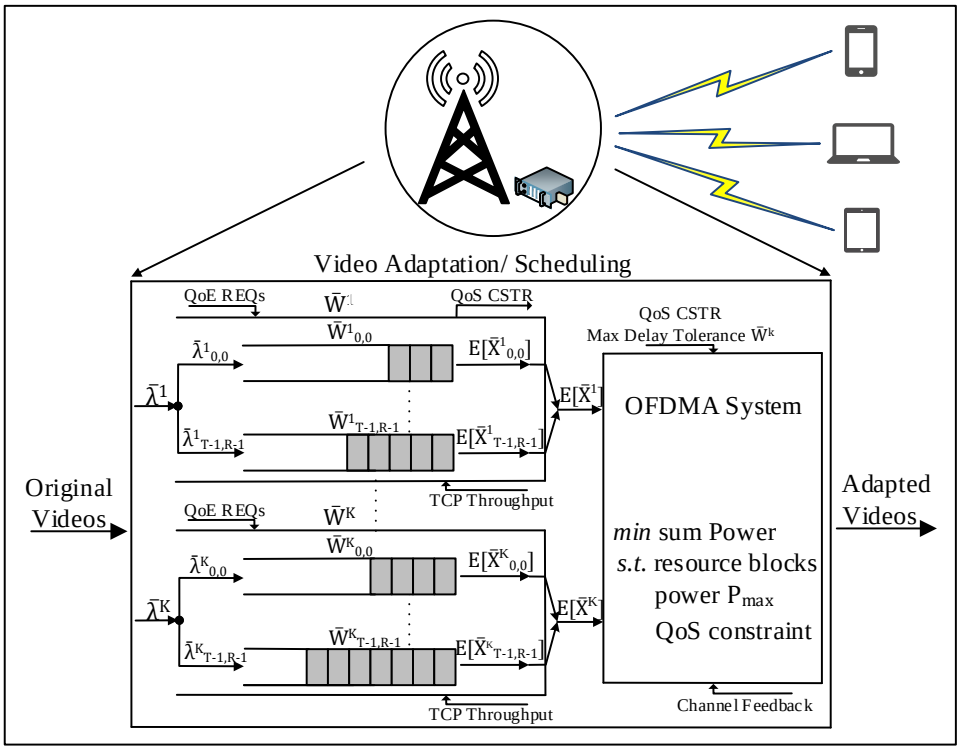


Fig. 4.1: Video adaptation/ scheduling system at network edge.

increases the capacity to serve more concurrent streams. In this chapter, we consider queuing delay due to its importance and effects on overall end-to-end delay and jitter.

The rest of this chapter is organized as follows. The system model and problem formulation are presented in Sections 4.2 and 4.3, respectively. Section 4.3 also describes the proposed queuing-based video adaptation and resource allocation algorithm. Section 4.4 conducts numerical and simulation analyses of the model. The conclusion is presented in Section 4.5.

4.2 System Model

This chapter focuses on the downlink of LTE networks and considers a single-cell multi-user scenario as shown in Fig. 4.1. The system consists of K mobile users (video streams) indexed by the set $\mathcal{K} \triangleq \{1, \ldots, k, \ldots, K\}$, sharing L RBs indexed by $\mathcal{L} \triangleq \{1, \ldots, l, \ldots, L\}$ in an OFDMA cell. The channel is assumed

to be frequency-selective Rayleigh fading, with flat fading within each RB. Each H.264/SVC stream has a number of temporal layers and quality layers.Temporal and quality layers of stream k are indexed by $\mathcal{T} \triangleq \{0, \ldots, t, \ldots, T^k - 1\}$ and $\mathcal{R} \triangleq \{0, \ldots, r, \ldots, R^k - 1\}$, respectively.

A statistical queuing model is deployed to express the delay limitation of a stream with an equivalent cross-layer constraint. Therefore, as in [101], it is assumed that packets arrive to each user k's buffer q^k based on a Poisson arrival process. Within q^k, the system places packets from the r^{th} quality layer of temporal layer t of sequence k into virtual queue (VQ) $q^k_{t,r}$, which follows the dynamics of M/G/1 queues [102]. The arrival rates in M/G/1 are Poisson processes, which are highly suitable for modeling SVC video traffic [103]. Moreover, the service time can follow any general statistical distribution. This is due to the fading channel, which makes the service process hard to model [101]. As shown in Fig. 4.1, we describe the parameters of q^k and $q^k_{t,r}$ by characteristic tuples $[\overline{\lambda}^k, \mathbb{E}[X^k], \overline{W}^k]$ and $[\overline{\lambda}^k_{t,r}, \mathbb{E}[X^k_{t,r}], \overline{W}^k_{t,r}]$, respectively. $\overline{\lambda}^k_{t,r}$, $\mathbb{E}\left[X^k_{t,r}\right]$ and $\overline{W}^k_{t,r}$ are the arrival rate, service time and waiting time of the packets at $q^k_{t,r}$, and $\overline{\lambda}^k$, $\mathbb{E}[X^k]$ and $\overline{W}^k$ are those of the packets at q^k.

Dropping packets from the VQs decreases the queuing delay and congestion in the network. It also increases the network capacity (the number of concurrent video requests that can be served). However, packet loss causes a certain reduction in the user QoE of a video depending on the importance of the video layer containing the dropped packet. This is estimated using the QoE metric proposed in Section 4.2.1. Thus, packets are dropped from different layers of a stream and a lower bit rate stream that satisfies the user's QoE requirements is produced.

The video adaptation problem is formulated as minimization of the queuing delay of user streams by means of dropping packets under QoE provisioning. The power-efficient RA OFDMA module then uses the calculated optimal queuing delay (which takes the QoE requirements and decoding deadline of the videos

into account) as a constraint that specifies the maximum delay tolerance for the videos. The OFDMA module transforms this delay constraint into a cross-layer constraint for OFDMA systems using the method proposed in Section 4.2.3 and finds the optimal RB and transmit power allocation policies $\mathbf{P}^*$ and $\mathbf{x}^*$, respectively to satisfy this constraint.

A summary of commonly used notation is provided in TABLE 4.1.

In the following sections, the proposed QoE metric model is explained. It provides a relationship between packet loss ratio and reduction in QoE. A relationship between the loss ratio at a queue and queuing delay is then formulated. This leads to a relationship between user QoE and packet loss ratio. Next, the queuing delay requirements is transformed into a cross-layer constraint by formulating a relationship between the average data rate of a stream and its delay threshold. Lastly, the video adaptation and RA problems are formulated.

4.2.1 QoE Metric Model

This chapter use the multi-scale structural similarity (MS-SSIM) index [104], which provides a good approximation of user-perceived quality. It calculates relative quality scores between a reference video frame and a distorted version. the QoS-QoE mapping technique proposed in [105] is deployed. It interprets packet loss ratio into a system-level QoE measure. The degradation in QoE caused by data drops at each video layer is calulated. Thus, for a given video stream, a Monte Carlo simulation is performed where a fixed percentage ($\rho^k_{t,r}$) of packets from each temporal/quality layer uniformly is dropped at random. The average QoE achievable $\mathbb{E}[q(\rho^k_{t,r})]$ when the packet loss ratio in a temporal/quality layer is $\rho^k_{t,r}$ is estimated. At each run, the video is decoded and the quality index is measured. For each $\rho^k_{t,r}$ value, different instances of the test are performed to find the average quality for $0 \leq \rho^k_{t,r} \leq 1$. Repeating over all temporal/quality layers, the empirical mapping is obtained.

TABLE 4.1: Commonly Used Notations

Notation	Description
K	Number of video users
T^k	Number of temporal layers of stream k
L	Number of RBs
R^k	Number of quality layers of stream k
q^k	User k's buffer
$q^k_{t,r}$	Virtual queue for packets of temporal/quality layer (t, r) of stream k
$\mathcal{D}^k$	Overall QoE reduction at stream k
$\overline{\lambda}^k_{t,r}$	Arrival rate of the packets at $q^k_{t,r}$
$\mathbb{E}[X^k_{t,r}]$	Service time of the packets at $q^k_{t,r}$
$\overline{W}^k_{t,r}$	Waiting time of the packets at $q^k_{t,r}$
$\rho^k_{t,r}$	Packet loss ratio at $q^k_{t,r}$
$\overline{\lambda}^k$	Arrival rate of the packets at q^k_k
$\mathbb{E}[X^k]$	Service time of the packets at q^k
$\overline{W}^k$	Waiting time of the packets at q^k
ρ^k	Stream k's packet loss ratio of
B	RB bandwidth
$\mathcal{P}_{k,l}$	Transmit power of user k over l^{th} RB
$x_{k,l}$	RB indicator vector

Proposition 2. *QoE reduction at stream k is defined as [105]*

$$\mathcal{D}^k = (1 - b_1^k) q_{\max}^k = \sum_{t=0}^{T^k-1} \sum_{r=0}^{R^k-1} \mathcal{D}(\rho_{t,r}^k), \tag{4.1}$$

where $q_{\max}^k$ is the quality in the absence of losses for stream k, b_1^k is the fractional quality degradation due to packet loss, and $\mathcal{D}(\rho_{t,r}^k) = q_{\max}^k - \mathbb{E}[q(\rho_{t,r}^k)]$ is the QoE degradation caused by packet loss ratio $\rho_{t,r}^k$ in temporal layer t, quality layer r.

Proof. The proof directly extends from [105]. Due to inter-layer dependencies between the quality layers in a SVC video stream, losses in base quality layers result in considerably higher quality degradation than losses in quality enhancement layers. (t_2, r_2) denotes a video layer which depends on layer (t_1, r_1) as the latter is used to reconstruct the former. S_1 represents a set of lost slices in (t_1, r_1) due to a packet loss ratio ρ_{t_1,r_1}. The degradation in video quality caused by losing S_1 is denoted by e_1. Moreover, S_2' is the set of slices in layer (t_2, r_2) affected by error propagation from the set S_1. Likewise, a set of lost slices caused by packet loss ratio ρ_{t_2,r_2} in (t_2, r_2) is denoted by S_2. e_2 denotes the degradation in video quality due to losing S_2. When the sets S_2' and S_2 are disjoint ($S_1 \cap S_2 = \varnothing$), the error propagation signal from layers (t_1, r_1) and (t_2, r_2) are independent, which results in the worst-case quality loss. Hence, the total quality loss sums exactly to $e_1 + e_2$. In the extreme case where $S_2 \cap S_2' = \varnothing$, the quality loss is $\mathbb{E}[e_1] + \mathbb{E}[e_2] = (q_{\max} - \mathbb{E}[q(\rho_{t_1,r_1})]) + (q_{\max} - \mathbb{E}[q(\rho_{t_2,r_2})])$, where $q_{\max}$ is the quality achieved in the absence of packet losses and $\mathbb{E}[q(\rho_{t,r})])$ is the average video quality achievable when the packet loss ratio is $\rho_{t,r}$. Due to the disjoint error propagation paths, the result generalizes to a set of packet loss ratios $\rho_{t,r}$ combined where the total quality reduction $\mathcal{D}$ is at most $\sum_{t=1}^{T} \sum_{r=1}^{R^k} [q_{\max} - \mathbb{E}[q(\rho_{t_2,r_2})])]$. Therefore, the reduction in video quality is upper bounded by

$$\mathcal{D} \leq \sum_{t=1}^{T} \sum_{r=1}^{R} [q_{\max} - \mathbb{E}[q(\rho_{t_2,r_2})])] \,. \tag{4.2}$$

Given the above upper bound on $\mathcal{D}$, it is adequate to select $\rho_{t,r}$ such that $\mathcal{D} = (1-\alpha)q_{\max} = \sum_{t=0}^{T}\sum_{r=0}^{R}[q_{\max} - \mathbb{E}[q(\rho_{t,r})])]$, where α denotes the fractional quality decrease caused by packet loss. ■

In case of TCP-based video streaming, the loss visibility of packets from each video layer over time is quantified using the ACK history, as in [105]. After a GoP is transmitted and its complete ACK history is fed back to the transmitter, a replica of the decoded GoP is reconstructed with the losses from each layer. Then, the corresponding packet loss is computed directly from the ACK history to estimate the channel distortion effects on each video layer.

4.2.2 MAC-Layer Modeling from a Cross-Layer Perspective

The average length of M/G/1 queue $q_{t,r}^k$ is given by [102]

$$\overline{L} = \frac{\overline{\lambda}^2 \mathbb{E}[X^2]}{2(1-\overline{\lambda}\mathbb{E}[X])}, \tag{4.3}$$

where $\overline{L}, \mathbb{E}[X], \mathbb{E}[X^2]$ and $\overline{\lambda}$ are used to denote $\overline{L}_{t,r}^k, \mathbb{E}[X_{t,r}^k]$, $\mathbb{E}[{X_{t,r}^k}^2]$ and $\overline{\lambda}_{t,r}^k$, respectively. $\mathbb{E}[X]$ and $\mathbb{E}[X^2]$ are the first and second moments of the service time at queue $q_{t,r}^k$. The average arrival rate of $q_{t,r}^k$ can be estimated by $\overline{\lambda}_{t,r}^k = \overline{s}_{t,r}^k (n_{t,r}^k / N^k) f^k$ [106], where $\overline{s}_{t,r}^k$ is the average size of a video frame in temporal layer t of the r^{th} quality layer, N^k is the number of frames in a GoP and f^k is the frame rate of stream k. $n_{t,r}^k$ is the number of frames in the t^{th} temporal layer of each quality layer, which can be derived from [107]

$$n_{t,r}^k = \begin{cases} 1 & \text{if } t \in \{0,1\} \\ 2^{t-1} & \text{if } 2 \le t \le \log_2 N^k. \end{cases} \tag{4.4}$$

Based on the Little theorem [102], the average waiting time in each queue is

$\overline{W}^k_{t,r} = \overline{L'}^k_{t,r}/\overline{\lambda}^k_{t,r}$, where $\overline{L'}^k_{t,r} = (1-\rho^k_{t,r})\overline{L}^k_{t,r}$ is the average queue length in the presence of packet loss ratio $\rho^k_{t,r}$ in $q^k_{t,r}$. Therefore, substituting (4.3) and $\overline{L'}^k_{t,r}$ into $\overline{W}^k_{t,r} = \frac{\overline{L'}^k_{t,r}}{\overline{\lambda}^k_{t,r}}$, the average waiting time in an M/G/1 queue is

$$\overline{W} = \frac{(1-\rho)\overline{\lambda}\mathbb{E}\left[X^2\right]}{2(1-\overline{\lambda}\mathbb{E}\left[X\right])}, \tag{4.5}$$

where $\overline{W}$ and ρ are used to denote $\overline{W}^k_{t,r}$ and $\rho^k_{t,r}$, respectively.

4.2.3 Delay Requirements to Data Rate Transformation

The maximum delay tolerance $\overline{W}^k_{\max}$ is estimated in the next section, which puts an upper-bound on the delay experienced by stream k. However, in order to transform this QoS constraint into a cross-layer constraint, using an M/G/1 queuing model, *Proposition 2* formulates a relationship between the average scheduled effective data rate of each user k and $\overline{W}^k_{\max}$.

Proposition 3. *A necessary condition to meet a maximum delay of $\overline{W}^k_{\max}$ for a stream k in an OFDMA system is [101]*

$$\mathbb{E}\left[\sum_{l=1}^{L} R_{k,l}\cdot x_{k,l}\right] \geq \left(\sqrt{\overline{\lambda}^k\overline{W}^k_{\max}\left(\overline{\lambda}^k\overline{W}^k_{\max}-2\rho^k+2\right)} + \overline{\lambda}^k\overline{W}^k_{\max}\right)\frac{S}{2\cdot B\cdot t_s\cdot\overline{W}^k_{\max}}, \qquad \forall k\in\mathcal{K}, \tag{4.6}$$

where B is the bandwidth of each RB, t_s is the scheduling slot duration and S is the size of each packet. $\overline{\lambda}^k$ and ρ^k are the average arrival rate and packet loss ratio at q^k. $R_{k,l} = B\log_2\left(1+\frac{\mathcal{P}_{k,l}|h_{k,l}|^2}{\sigma^2}\right)$ is the upper bound on the achievable service rate for user k over RB l, where h^k_l is the channel fading coefficient and σ^2 denotes the noise power.

Proof. The proof extends from [101]. A necessary and sufficient condition for this

constraint is

$$\overline{W}^k = \frac{(1-\rho^k)\overline{\lambda}^k \mathbb{E}[X^{k^2}]}{2(1-\overline{\lambda}^k \mathbb{E}[X^k])} \leq \overline{W}^k_{\max}, \tag{4.7}$$

where $\overline{W}^k$ is the average delay in the k^{th} user's stream M/G/1 queue, $\overline{\lambda}^k$ is the average arrival rate of stream k, X^k the service time of the packets of stream k and $\mathbb{E}[X^k]$ represents the average service time with a second-order moment denoted by $\mathbb{E}[X^{k^2}]$. ρ^k is stream k's packet loss ratio, which is the optimal value of the variable in optimization problem 4.11. The maximum delay tolerance for stream k, $\overline{W}^k_{\max}$, is the optimal solution (objective function) of problem 4.11.

The first-order moment of service time X^k of user k's stream is defined as $\mathbb{E}[X^k] = \frac{S}{\mathbb{E}[b^k_l]}$ and the second-order moment as $\mathbb{E}[X^{k^2}] \geq \frac{S^2}{\mathbb{E}[b^{k^2}_l]}$, where b^k is the equivalent rate at queue q given as the number of bits loaded to L RBs, i.e., $b^k = \sum_{l=1}^{L} b^k_l$, where b^k_l is the number of identically distributed bits of the k^{th} user loaded to RB l. By substituting $\mathbb{E}[X^k]$ and $\mathbb{E}[X^{k^2}]$ into $\overline{W}^k$, the traffic arrival rate of stream k is given by

$$2\overline{W}^k_{\max} b^{k^2} - 2\overline{W}^k_{\max}\overline{\lambda}^k S b^k - \overline{\lambda}^k S^2 + \rho^k \overline{\lambda}^k S^2 = 0. \tag{4.8}$$

By performing the calculations above, b^k can be derived as

$$b^k = \frac{S}{2\overline{W}^k_{\max}} \left(\sqrt{\overline{\lambda}^k \overline{W}^k_{\max} \left(\overline{\lambda}^k \overline{W}^k_{\max} - 2\rho^k + 2 \right)} + \overline{\lambda}^k \overline{W}^k_{\max} \right), \qquad \forall l \in \mathcal{L}. \tag{4.9}$$

The above condition indicates that the number of bits of stream k loaded to RB $l \in \mathcal{L}$ is lower bounded according to the user's queuing characteristics. It represents the number of the arrival bits at the queue of each user's stream over all allocated RBs. Hence, taking the RB allocation index $x_{k,l}$, the duration of a time slot t_s, the RB bandwidth B into consideration, the user k's average effective

data rate at the cross-layer is

$$\mathbb{E}\left[\sum_{l=1}^{L} R_{k,l} \cdot x_{k,l}\right] \geq \frac{b^k}{B \cdot t_s}. \tag{4.10}$$

■

4.3 Video Adaptation and Resource Allocation

In this section, the video adaptation is first formulated as a QoE-constrained queuing delay minimization problem. The optimal packet loss ratio and service rate is derived, which minimize the queuing delay and adapt the stream based on the QoE constraint. The cross-layer RA problem is then formulated as a power minimization problem under the delay constraint derived in the video adaptation phase.

4.3.1 Optimization Based Video Adaptation/Scheduling

The objective is to maximize capacity, which is defined as the number of concurrent streams that can be served while meeting each stream's QoE and delay requirements. This is acheived by minimizing the average queuing delay of each stream, and hence decreasing the queue length in the buffer. This, in turn, provides a lower bit rate version of the stream by dropping packets, subject to

minimum QoE and maximum decoding deadline constraints at all VQs.

$$\min_{\rho,\mathbb{E}[X]} \mathbb{E}\left[\sum_{t=0}^{T^k-1}\sum_{r=0}^{R^k-1} \overline{W}_{t,r}^{k}\right] \tag{4.11}$$

suject to:

$$\overline{W}_{t,r}^{k} \leq \overline{W}_{\max,t,r}^{k} \qquad \forall k \in \mathcal{K}, \forall t \in \mathcal{T}, \forall r \in \mathcal{R} \tag{4.11a}$$

$$\mathcal{D}^k \leq \mathcal{D}_{\max}^k \qquad \forall k \in \mathcal{K} \tag{4.11b}$$

$$\sum_{t=0}^{T^k-1}\sum_{r=0}^{R^k-1} \mathbb{E}\left[X_{t,r}^k\right] \leq C^k \qquad \forall k \in \mathcal{K}. \tag{4.11c}$$

The objective function (4.11) minimizes the average queuing delay of video streams. Constraint (4.11a) ensures that the average waiting time in each VQ does not exceed the respective average expiry time (decoding deadline) $\overline{W}_{\max,t,r}^{k}$. A fixed structure of B pictures is assumed. Therefore, the average expiry time of packets in $q_{t,r}^k$ can be adequately approximated by $\overline{W}_{\max,t,r}^{k} \approx \frac{1}{f^k}$ [86]. Constraint (4.11b) means that QoE reduction at stream k does not exceed $\mathcal{D}_{\max}^k$, which is the maximum allowable degradation in the QoE of the stream (decided by operator). In ABS, the requested video rate is adapted to the user's TCP throughput. Therefore, (4.11c) ensures that sum of the average service rates of stream k's VQs is upper-bounded by the end-user's TCP throughput C^k.

4.3.2 Power-Efficient Delay-Constrained Resource Allocation

We deploy the cross-layer RA problem in [101]. It targets to minimize the power transmitted from the BS to K users while satisfying the delay limitation of each

stream derived in Section 4.3.1. The RA problem is formulated as

$$\min_{P,x} \mathbb{E}\left[\frac{1}{L}\sum_{k=1}^{K}\sum_{l=1}^{L}\mathcal{P}_{k,l}\cdot x_{k,l}\right] \tag{4.12}$$

suject to:

$$\mathbb{E}\left[\frac{1}{L}\sum_{k=1}^{K}\sum_{l=1}^{L}\mathcal{P}_{k,l}\cdot x_{k,l}\right] \leq P^{\max} \qquad \forall k \in \mathcal{K} \tag{4.12a}$$

$$\sum_{k=1}^{K} x_{k,l} \leq 1 \qquad \forall l \in \mathcal{L} \tag{4.12b}$$

$$x_{k,l} \in \{0,1\},\ \mathcal{P}_{k,l} \geq 0 \qquad \forall k \in \mathcal{K},\ \forall l \in \mathcal{L} \tag{4.12c}$$

$$\overline{W}^{k} \leq \overline{W}^{k}_{\max} \qquad \forall k \in \mathcal{K}. \tag{4.12d}$$

The objective of the optimization problem in (4.12) is power and RB allocation in order to minimize the total transmit power in the downlink. $P^{\max}$ in (4.12a) puts an upper limit on the average total available power at the BS. (4.12b) and (4.12c) indicate that each RB can be allocated to one receiver exclusively. Binary variables $x_{k,l}$ is used to represent the RB assignment. (4.12d) expresses the delay limitation of stream k. $\overline{W}^{k}_{\max}$ is the maximum delay tolerance for the k^{th} stream, where $\overline{W}^{k}_{\max} = \mathbb{E}[\sum_{t=0}^{T^k-1}\sum_{r=0}^{R^k-1}\overline{W}^{*k}_{t,r}], \forall k$. $\mathbb{E}[\sum_{t=0}^{T^k-1}\sum_{r=0}^{R^k-1}\overline{W}^{*k}_{t,r}]$ is the optimal solution of problem (4.11) for stream k. (4.6) provides a necessary condition for constraint (4.12d).

A summary of the proposed algorithm is provided in Algorithm 2.

4.4 Numerical and Simulation Results

As in Chapter 3, JSVM 9.19.15 [100] is used to encode/decode the SVC streams. The video sequences"city" (bit rate $\sim$ 450 kbps) and "foreman" (bit rate $\sim$ 400 kbps) are used in the simulations. The maximum frame rate is 30 fps and the number of temporal layers and quality enhancement layers are both set to

Algorithm 2: Proposed video adaptation/ scheduling algorithm

1: Given K streams with properties $T^k, R^k, f^k, N^k, \overline{\lambda}^k$ and maximum allowable QoE degradation $\mathcal{D}^k_{\max}$;

2: Use Monte Carlo simulations to estimate $\mathcal{D}(\rho^k_{t,r})$ for each stream k for $0 \leq \rho^k_{t,r} \leq 1, \forall t \in \mathcal{T}, \forall r \in \mathcal{R}$;

3: Solve (4.11) to find the optimal packet loss ratio ${\rho^*}^k_{t,r}$ of stream k's temporal/quality layers which produces a lower-rate stream based on $\mathcal{D}^k_{\max}$;

4: Obtain the optimal queuing delay ${\overline{W}^*}^k_{t,r}$ from (4.11), which takes the QoE requirements and decoding deadlines of temporal/quality layers of each stream k into account;

5: Calculate the maximum delay tolerance $\overline{W}^k_{\max} = \mathbb{E}[\sum_{t=0}^{T^k-1} \sum_{r=0}^{R^k-1} {\overline{W}^*}^k_{t,r}]$ and optimal ${\rho^*}^k$ for stream k;

6: Transform (4.12d) to a cross-layer constraint using (4.6);

7: Solve (4.12) to derive the optimal power ${P^*}^k_l$ and RB assignment ${x^*}^k_l$ under maximum delay tolerance constraint;

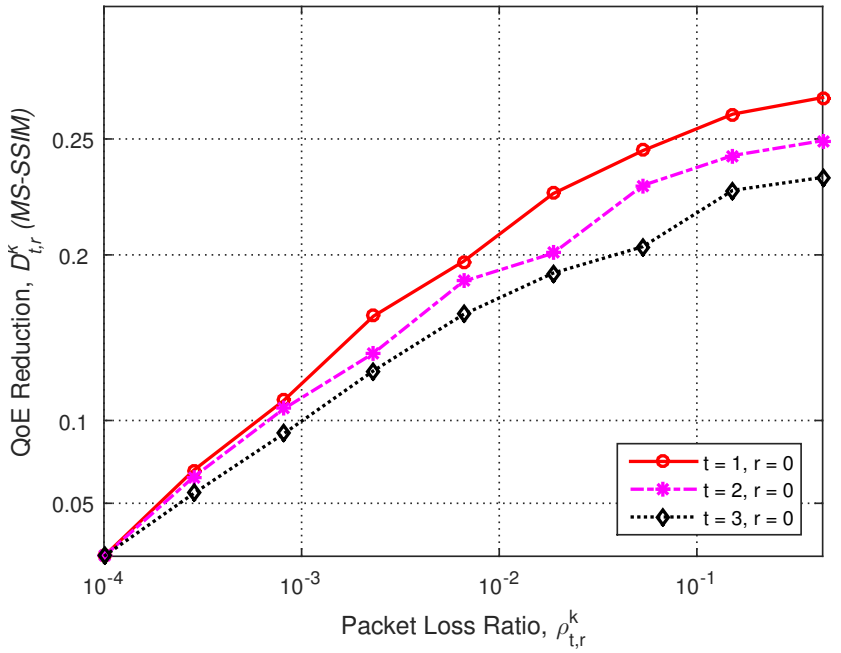
4. Using the method in Section 4.2.1, the loss visibility of packets from each video layer is estimated. Fig. 4.2 shows the QoE reduction of "city" sequence (which involves more background motion) when a uniform packet loss is applied to each layer. As shown in Fig. 4.2, losses in layers with layer identifier $r = 0$ result in significant degradation in video quality. Due to packet scalability, quality degradation has considerably lower severity when losses occur in upper temporal/quality layers [105].

We now consider the downlink of a single-cell OFDMA system. The bandwidth is 10 MHz (50 usable RBs per TTI). The channel model accounts for Rayleigh fading, large scale path loss and log-normal shadowing. The noise power is -174 dBm/Hz. 8 uniformly distributed users with a minimum distance of 50 m from the eNodeB are assumed.

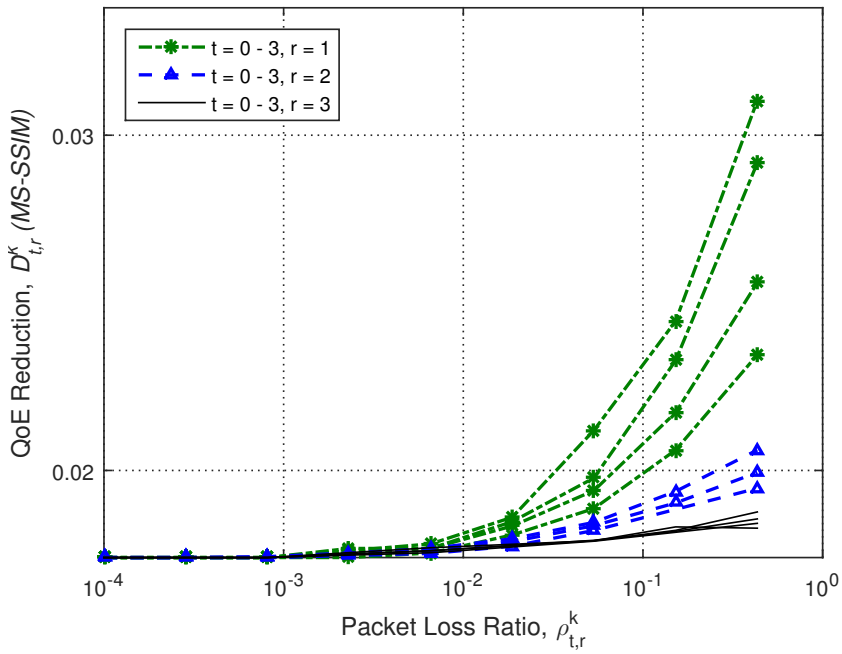
Two scenarios are consider, in each of which, users have different QoE requirements. In *Scenario 1*, "foreman" video streams are transmitted to the users and each video is adapted dynamically based on the maximum allowable QoE degradation $\mathcal{D}^k_{\max} = 0.3$. In *Scenario 2*, which has higher QoE requirements, we transmit "city" streams and set $\mathcal{D}^k_{\max}$ to 0.1.

The proposed algorithm is compared with WSPmin [87] and VAWS [8] RA schemes. WSPmin minimizes the total transmit power with a minimum rate constraint. In VAWS, RBs are assigned to satisfy minimum rate constraint with the assumption of equal power allocation per RB. It then refines the initial uniform power allocation to ensure that minimum rate requirements are met. It repeats the previous phases to refine power allocation.

The data rate requirements for the users served by WSPmin and VAWS are randomly varying from 100 kbps to 400 kbps as multiples of 50 kbps. Fig. 4.3 demonstrates the CDF of sum power for different RA schemes generated over 100 iterations using MATLAB [108]. It is noted that in *Scenario 1*, the proposed scheme outperforms both WSPmin and VAWS algorithms in terms of power ef-



(a)



(b)

Fig. 4.2: QoE reduction vs. packet loss ratio for "city" sequence with y-axis in log scale: (a) base layers (t = 1 to 3, r=0); (b) enhancement layers (t=0 to 3, r=1).

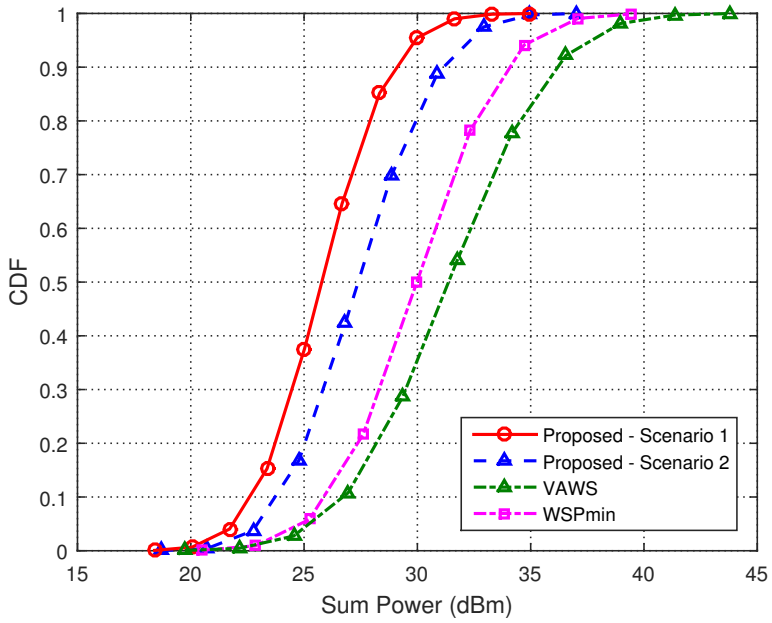


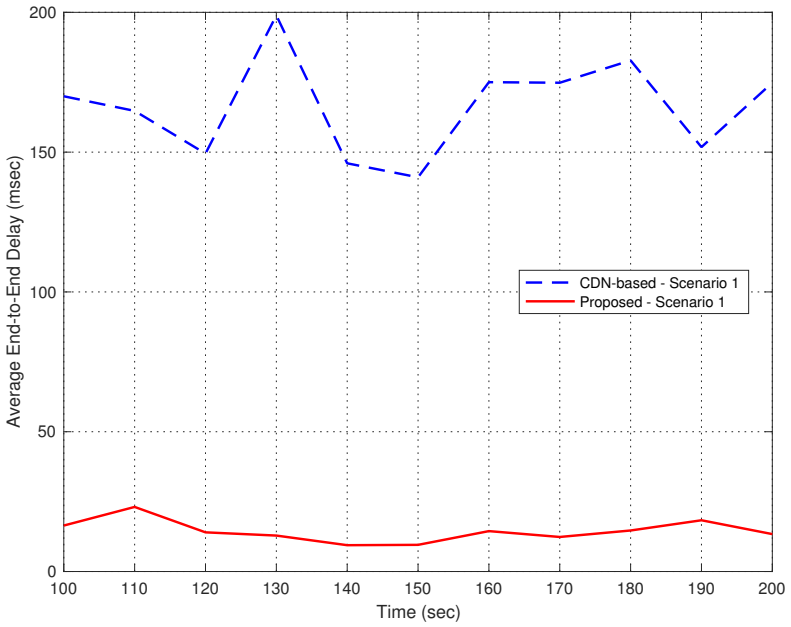
Fig. 4.3: CDF of sum power.

ficiency by performing 17.29% better than the former and 24.7% better than the latter in 90% of the times. Likewise, in *Scenario 2*, compared with WSPmin and VAWS, the proposed approach results in 12.37% and 19.81% power-efficiency improvement in 90% of the times, respectively.

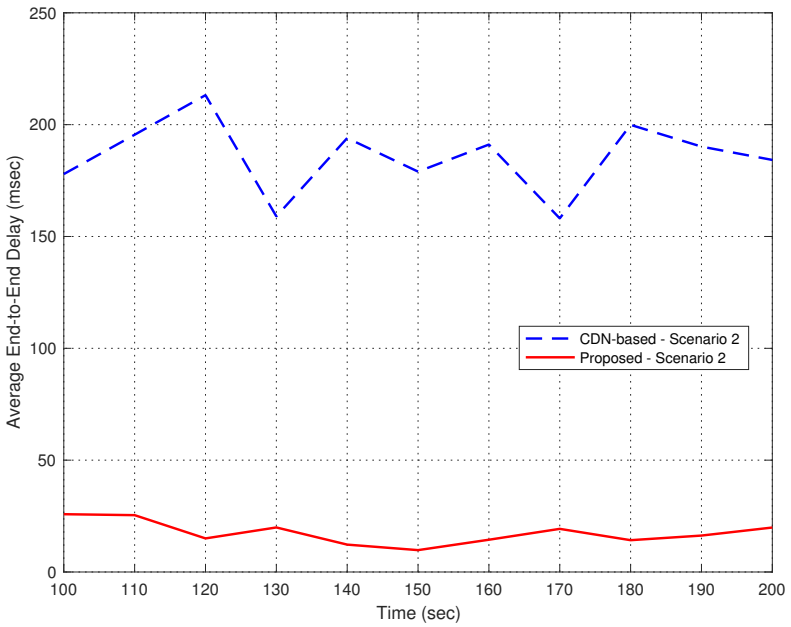
Fig. 4.4 shows a comparison of the proposed approach and the widely used CDN-based ABS in terms of end-to-end delay using OPNET [109]. Compared with CDN-based streaming where "foreman" videos with $\mathcal{D}^k_{\max}$ set to 0.3 (*Scenario 1*) and "city" videos with $\mathcal{D}^k_{\max} = 0.1$ (*Scenario 2*) are transmitted to users, the proposed scheme decreases delay by 89.26% and 86.44%, respectively.

4.5 Conclusion

This chapter has proposed a queuing-based in-network video adaptation and RA scheme. This is tailored for SVC video contents cached at the mobile edge. Therefore, by selectively dropping packets from a video stream, a lower bit rate is produced, which reduces delay and satisfies a target user QoE. Resources are



(a)



(b)

Fig. 4.4: Comparison of end-to-end delay: (a) *Scenario 1*; (b) *Scenario 2*.

then allocated to meet the delay limitation of the lower rate stream. The results show that the proposed approach achieves significant performance improvement in terms of reducing delay and power consumption.

Chapters 3 and 4 have made the assumption that a reactive caching technique is in place and performed in-network video adaption for the reactively cached contents. A proactive SVC video caching approach is proposed in the next chapter to increase the video capacity of the wireless network, and reduce network load and latency. In the next chapter, it is assumed that that video adaptation is carried out using DASH rate adaptation mechanism.

Chapter 5

Cost-Effective Driven Mobile Video Caching

5.1 Introduction

Chapters 3 and 4 proposed two different in-network video adaptation schemes for video contents that are reactively cached at the edge of the network. A different approach to bring content closer to the end user would be to proactively cache ABS videos.

With in-network caching, users can access popular content from caches of nearby MNO gateways [i.e. EPC and RAN] [10–15], therefore significantly reducing video str eaming latency. Furthermore, from the Internet service providers (ISP)'s perspective, in-network caching also helps to reduce inter- and intra-ISP traffic and, so, to optimize operating costs for leasing expensive fiber lines that connect eNodeBs to EPC [14, 15].

Several approaches have been proposed to analyze intelligent caching strategies for mobile content caching inside MNO's network [12–14]. An extensive overview of the techniques for in-network content caching in 5G mobile networks has been introduced in [15], whereas different proactive mobile caching schemes

have been discussed in [7, 10]. The current chapter contributes to this stream of work by analyzing the trade-off between the potential savings from- and infrastructural costs of hierarchical in-network caching.

5.2 Contributions and Outline

The main contributions of this chapter can be summarized as follows:

- this chapter presents the first attempt to formulate the problem of storage provisioning for a hierarchical in-network video caching which optimizes the trade-off between the cost of transmission bandwidth and the cost of storage.
- the focuse of this chapter is on SVC-based DASH format, which encodes a video into different quality layers and is therefore more resource-efficient than traditional H.264/AVC-based DASH in which a separate AVC video file is encoded for each video quality format [2].
- the storage provisioning problem formulated in this chapter is solved using CDT [26]. More specifically, the proposed BIP problem is formulated into a canonical dual problem in continuous space, which is a concave maximization problem. Additionally, the conditions under which the solutions of the canonical dual problem and primal problem are identical is provided.
- The canonical dual problem results in complex non-linear equations which are efficiently solved by applying IWO algorithm [27].

The rest of the chapter is structured as follows. Section 5.3 describes the system model. The cache provisioning problem is formulated in 5.4. Section 5.5 presents the canonical dual framework. Section 5.6 conducts a simulation analysis of the model. This chapter is finalized by a conclusion in Section 5.7.

5.3 System Model

The system consists of I video streams, which are indexed by the set $\mathcal{I} \triangleq \{1, \ldots, i, \ldots, I\}$. Different quality layers of a video stream is indexed by the set $\mathcal{J} \triangleq \{1, \ldots, j, \ldots, J\}$. $q_{i,j}$ denotes the j^{th} quality layer of video i, which has a size and popularity (hit rate) of $f_{i,j}$ and $p_{i,j}$, respectively. A hierarchical in-network caching system is considered with caches within different levels, as shown in Fig. 5.1. Different levels of the hierarchical architecture are indexed by $\mathcal{N} \triangleq \{1, \ldots, n, \ldots, N\}$. One example of a hierarchical in-network caching system can be found in [12], which defines a cache hierarchy tree of three levels with first, second and third level nodes being eNodeBs, S-GWs and P-GW, respectively. More examples can be found in [14].

5.3.1 Notations and Variables

Cache Assignment Binary Decision Variable ($x_{n,i,j} \in \{0, 1\}$)

$x_{n,i,j}$ represents the cache assignment for $q_{i,j}$ in the n^{th} cache hierarchy, where $x_{n,i,j} = 1$ indicates that an storage size of $f_{i,j}$ should be assigned to a cache in level n of the hierarchical in-network caching system and $x_{n,i,j} = 0$ otherwise.

Provisioned Storage Size (s_n)

s_n denotes the storage capacity that is required to be assigned to the n^{th} level of the in-network hierarchy.

Maximum Possible Storage Size (m_n)

m_n is the maximum possible storage capacity that the MNO can install on the n^{th} level of hierarchical caching system.

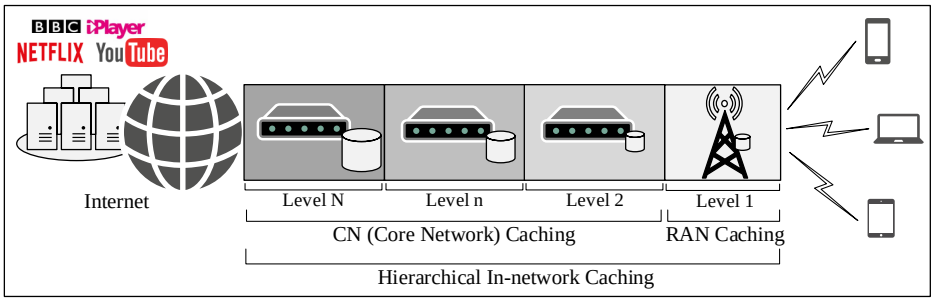


Fig. 5.1: A hierarchical in-network video caching system.

Offloaded Traffic ($l_{n,i,j}$)

$l_{n,i,j}$ represents the reduction in the transmission bandwidth as a result of caching $q_{i,j}$ in level n of the in-network caching hierarchy, where $l_{n,i,j} = f_{i,j} \times p_{i,j} \times x_{n,i,j}$.

Return Function ($\mathcal{R}_n$)

It is assumed that the benefit of transmission bandwidth saving follows a predefined function $\Gamma : \mathbb{R} \rightarrow \mathbb{R}$. Thus, the benefit derived from the reduction in transmission bandwidth when videos are cached in the n^{th} level of the in-network caching hierarchy is estimated as

$$\mathcal{R}_n\left(l_{n,i,j}\right) = \Gamma\left(\sum_{i=1}^{I}\sum_{j=1}^{J} l_{n,i,j}\right) \qquad \forall n \in \mathcal{N}. \tag{5.1}$$

Cost Function ($\mathcal{C}_n$)

It is assumed that the cache storage cost follows a predefined function $\Lambda : \mathbb{R} \rightarrow \mathbb{R}$. Hence, the cost associated with provisioned storage size s_n is

$$\mathcal{C}_n\left(s_n\right) = \Lambda\left(s_n\right) \qquad \forall n \in \mathcal{N}. \tag{5.2}$$

Both return and cost functions can be any appropriate function defined by the MNO. However, without loss of generality, it may be assumed that they are either linear or logarithmic.

5.4 Problem Formulation

The cache provisioning problem is formulated as follows.

$$\max_{\mathbf{x}} \frac{\sum_{n=1}^{N} \mathcal{R}_n(l_{n,i,j})}{\sum_{n=1}^{N} \mathcal{C}_n(s_n)} \tag{5.3}$$

subject to:

$$s_n = \sum_{i=1}^{I} \sum_{j=1}^{J} f_{i,j} x_{n,i,j} \leq m_n \qquad \forall n \in \mathcal{N} \tag{5.3a}$$

$$\sum_{n=1}^{N} x_{n,i,j} \leq 1 \qquad \forall i \in \mathcal{I}, \forall j \in \mathcal{J} \tag{5.3b}$$

$$x_{nij-1} \geq x_{n,i,j} \qquad \forall n \in \mathcal{N}, \forall i \in \mathcal{I}, \forall j \in \mathcal{J} - \{1\} \tag{5.3c}$$

$$x_{n,i,j} \in \{0,1\} \qquad \forall i \in \mathcal{I}, \forall j \in \mathcal{J}, \forall n \in \mathcal{N}. \tag{5.3d}$$

The objective of optimization problem (5.3) is to find the optimal provisioned storage capacity, s_n, which maximizes the return on investment, defined as the ratio of overall return (5.1) to overall cost (5.2). Constraint (5.3a) ensures that the cache storage allocated to the n^{th} level of the hierarchical caching system is upper-bounded by the maximum possible storage capacity threshold, m_n. Constraint (5.3b) indicates that each video can be cached in one hierarchical levele inside the in-network caching architecture exclusively. Constraint (5.3c) ensures that if a video quality layer is cached, all the lower quality layers are cached too. Binary variables $x_{n,i,j} \in \{0,1\}$ are used, as explained in section 5.3.1.

The optimization problem (5.3) is difficult to solve due to its combinatorial nature. As an intermediate step towards solution, (5.3) is converted into a BIP problem by defining a cache allocation matrix, where instead of making decisions on the basis of individual video quality layer, decisions are made on the basis of feasible set of video layer cache allocation patterns that satisfies constraint (5.3c). The idea of pattern allocation is similar to [110]. All the combinations of video streams and the respective quality layers are indexed by the set $\mathcal{K} \triangleq$

$\{1, \ldots, k, \ldots, K\}$. $K = |\mathcal{K}|$ denotes the cardinality of set $\mathcal{K}$. The cache allocation matrix is of the order $K \times A$, where each row corresponds to the video stream-video quality layer combination index and each column corresponds to a feasible cache allocation pattern [meeting constrain (5.3c)]. A denotes the total number of feasible allocation patterns. The basic idea of this cache allocation matrix, for the case of 2 videos each with 2 quality layers is illustrated by (5.4). In any allocation pattern (i.e., any column), a "1" is placed when the video quality layer is cached, otherwise a "0" is placed.

$$\mathbf{Y}^n = \begin{bmatrix} 1 & 1 & 0 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 0 & 1 \end{bmatrix} \tag{5.4}$$

$\mathbf{x} \triangleq [\mathbf{x}_n]_{N\times 1}$ is defined as a cache indicator vector, where $\mathbf{x}_n = [x_{n,a}]_{A\times 1}$. Each entry $x_{n,a} \in \{0, 1\}$ indicates whether the cache allocation pattern a is allocated to hierarchical caching level n or not.

Note that all the caches in the hierarchical in-network caching system have the

same allocation patterns matrix. (5.3) is rewritten as a BIP problem as follows:

$$\min_{\mathbf{x}} \left\{ \mathcal{P}(\mathbf{x}) = -\frac{\sum_{n=1}^{N}\sum_{a=1}^{A} \mathcal{R}_{n,a} x_{n,a}}{\sum_{n=1}^{N}\sum_{a=1}^{A} \mathcal{C}_{n,a} x_{n,a}} \right\} \tag{5.5}$$

subject to:

$$\sum_{a=1}^{A} s_{n,a} x_{n,a} \leq m_n \qquad \forall n \in \mathcal{N} \tag{5.5a}$$

$$\sum_{n=1}^{N}\sum_{a=1}^{A} Y_{k,a}^{n} x_{n,a} \leq 1 \qquad \forall k \tag{5.5b}$$

$$x_{n,a}\left(x_{n,a} - 1\right) = 0 \qquad \forall n \in \mathcal{N}, \forall a \tag{5.5c}$$

$$\sum_{a=1}^{A} x_{n,a} = 1 \qquad \forall n \in \mathcal{N}. \tag{5.5d}$$

where $\mathcal{R}_{n,a}$ and $\mathcal{C}_{n,a}$ are the transmission bandwidth benefit and storage cost of allocating pattern a to hierarchical cache level n, which results in a provisioned storage size of $s_{n,a}$. For a cache level n, constraint (5.5a) puts an upper-bound of m_n on the provisioned storage size, which is equivalent to constraint (5.3a). Constraint (5.5b) ensures the exclusivity of the allocated videos, where $Y_{k,a}^{n}$ denotes the k^{th} row and a^{th} column of the matrix $\mathbf{Y}^n$, where $Y_{k,a}^{n} = 1$ indicates that the video stream-video quality layer combination k should be cached in hierarchical level n and $Y_{k,a}^{n} = 0$ otherwise. Constraint (5.5c) is a pure binary constraint that ensures $x_{n,a} \in \{0, 1\}$. Constraint (5.5d) ensures that at most one allocation pattern is chosen for each caching level.

Although the optimization problem (5.5) is simpler and more tractable than (5.3), the solution is still exponentially complex.

5.5 Canonical Dual Framework

5.5.1 Dual Problem Formulation

The BIP problem (5.5) is converted into a continuous space canonical dual problem using CDT [26,111,112], which is solved in continuous space. The conditions under which the solution of the canonical dual problem is identical to that of the primal is then identified. A generic framework for solving 0-1 quadratic problems using CDT can be found in [113]. However, due to additional constraints, the proposed problem here is more complex. A framework for solving resource allocation BIP and mixed integer programming (MIP) problems using CDT is given in [114] and [112], which will be extended to solve (5.5).

The feasible space for the primal problem (5.5) is defined by $\mathcal{X}_p = \{\mathbf{x} \in \{0,1\}^{NA}\}$. The equality constraints (5.5c) and (5.5d) are temporarily relaxed to inequalities and the primal problem with these inequality constraints are transformed into continuous domain canonical dual problem. The problem is then solved in continuous space and the conditions under which the solutions of the canonical dual problem and primal problem are identical are provided .

As a key step towards canonical dual formulation, the geometrical operator for the primal problem is defined as $\wedge(\mathbf{y}) = (\boldsymbol{\delta}, \boldsymbol{\beta}, \boldsymbol{\tau}, \boldsymbol{\sigma}) \in \mathcal{Y}_g$, which is a vector valued mapping where $\mathcal{Y}_g$ is the feasible space for $\mathbf{y}$, and

$$\begin{cases} \boldsymbol{\delta} = [\sum_{a=1}^{A} s_{n,a} x_{n,a} - m_n]_{N\times 1} \\ \boldsymbol{\beta} = [\sum_{n=1}^{N} \sum_{a=1}^{A} Y_{k,a}^{n} x_{n,a} - 1]_{K\times 1} \\ \boldsymbol{\tau} = [x_{n,a}\,(x_{n,a} - 1)]_{NA\times 1} \\ \boldsymbol{\sigma} = [\sum_{a=1}^{A} x_{n,a} - 1]_{N\times 1} \end{cases} \tag{5.6}$$

Therefore, the feasible space for $\mathbf{y}$ is defined by $\mathcal{Y}_g = \mathbb{R}^N \times \mathbb{R}^K \times \mathbb{R}^{NA} \times R^N | \boldsymbol{\delta} \leq 0, \boldsymbol{\beta} \leq 0, \boldsymbol{\tau} \leq 0, \boldsymbol{\sigma} \leq 0$.

Next, the indicator function is defined [113] as

$$V(\mathbf{y}) = \begin{cases} 0 & \text{if } \mathbf{y} \leq 0 \\ +\infty & \text{otherwise.} \end{cases} \tag{5.7}$$

The primal problem (5.5) is rewritten in the canonical form using indicator function (5.7) as follows:

$$\min \left\{ V(\wedge(\mathbf{y})) + \mathcal{P}(\mathbf{x}) \right\}. \tag{5.8}$$

$\mathbf{y}^* = (\boldsymbol{\delta}^*, \boldsymbol{\beta}^*, \boldsymbol{\tau}^*, \boldsymbol{\sigma}^*)$ is defined as the vector of dual variables associated with the corresponding restrictions $\mathbf{y} \leq 0$. The feasible space for $\mathbf{y}^*$ is defined by $\mathcal{Y}_d = \mathbb{R}^N \times \mathbb{R}^K \times \mathbb{R}^{NA} \times R^N | \boldsymbol{\delta}^* \geq 0, \boldsymbol{\beta}^* \geq 0, \boldsymbol{\tau}^* \geq 0, \boldsymbol{\sigma}^* \geq 0$. Based on the Fechnel transformation, the canonical sup-conjugate function of $V(\mathbf{y})$ is defined as

$$\begin{aligned} V^*(\mathbf{y}^*) &= \sup \left\{ \langle \mathbf{y}, \mathbf{y}^* \rangle - V(\mathbf{y}) | \mathbf{y} \in \mathcal{Y}_g, \mathbf{y}^* \in \mathcal{Y}_d \right\} \\ &= \sup_{\mathbf{y}^*} \left\{ \langle \boldsymbol{\delta}^T \boldsymbol{\delta}^* + \boldsymbol{\beta}^T \boldsymbol{\beta}^* + \boldsymbol{\tau}^T \boldsymbol{\tau}^* + \boldsymbol{\sigma}^T \boldsymbol{\sigma}^* - \mathcal{Y}_g \rangle \right\} \\ &= \begin{cases} 0 & \text{if } \boldsymbol{\delta}^*, \boldsymbol{\beta}^*, \boldsymbol{\tau}^*, \boldsymbol{\sigma}^* \geq 0 \\ +\infty & \text{otherwise.} \end{cases} \end{aligned} \tag{5.9}$$

Using the definition of sub-differential, it can be easily verified that if $\mathbf{y}^* > 0$, then the condition $\mathbf{y}^T \mathbf{y}^* = 0$ leads to $\mathbf{y} = 0$, and consequently $\mathbf{x} \in \mathcal{X}_p$. Hence, the dual feasible space for the primal problem in (5.5) is an open positive cone defined by $\mathcal{X}_p^\sharp = \{ \mathbf{y}^* \in \mathcal{Y}_d | \mathbf{y}^* > 0 \}$.

The total complementarity function [26] is defined as

$$\Xi(\mathbf{x}, \mathbf{y}^*) = \wedge(\mathbf{x})^T \mathbf{y}^* - V^*(\mathbf{y}^*) + \mathcal{P}(\mathbf{x}), \tag{5.10}$$

which is obtained by replacing $V(\mathbf{y}) = \wedge(\mathbf{x})^T \mathbf{y}^* - V^*(\mathbf{y}^*)$ (Fechnel-Young equal-

ity) in (5.8). The definitions of $\wedge(\mathbf{x}), V^*(\mathbf{y}^*)$ and $\mathcal{P}(\mathbf{x})$ are used to express $\Xi(\mathbf{x}, \mathbf{y}^*) = \Xi(\mathbf{x}, \boldsymbol{\delta}^*, \boldsymbol{\beta}^*, \boldsymbol{\tau}^*, \boldsymbol{\sigma}^*)$ as given by

$$\begin{aligned}\Xi(\mathbf{x}, \mathbf{y}^*) = \sum_{n=1}^{N}\sum_{a=1}^{A} x_{n,a}\Phi - \frac{\sum_{n=1}^{N}\sum_{a=1}^{A} \mathcal{R}_{n,a}x_{n,a}}{\sum_{n=1}^{N}\sum_{a=1}^{A} \mathcal{C}_{n,a}x_{n,a}} \\ - \sum_{n=1}^{N} \delta_n^* m_n - \sum_{k=1}^{K} \beta_k^* - \sum_{n=1}^{N} \sigma_n^* + \sum_{n=1}^{N}\sum_{a=1}^{A} \tau_{n,a}^* x_{n,a}^2,\end{aligned} \tag{5.11}$$

where $\Phi = \sum_{k=1}^{K} \beta_k^* Y_{k,a}^n + \delta_n^* s_{n,a} + \sigma_n^* - \tau_{n,a}^*$. Next, the canonical dual function [26, 113] is defined using the canonical dual variables as

$$\Upsilon(\boldsymbol{\delta}^*, \boldsymbol{\beta}^*, \boldsymbol{\tau}^*, \boldsymbol{\sigma}^*) = \text{sta}\left\{\Xi(\mathbf{x}, \boldsymbol{\delta}^*, \boldsymbol{\beta}^*, \boldsymbol{\tau}^*, \boldsymbol{\sigma}^*)\right\}, \tag{5.12}$$

where sta$(\cdot)$ denotes finding the stationary point of the function. The stationary point of $\Xi(\mathbf{x}, \mathbf{y}^*)$ occurs at

$$x_{n,a}(\mathbf{y}^*) = \frac{1}{2} - \frac{1}{2\tau_{n,a}^*}\left(\sum_{k=1}^{K} \beta_k^* Y_{k,a}^n + \delta_n^* s_{n,a} + \sigma_n^*\right) \qquad \forall n, a, \tag{5.13}$$

where the stationary point is obtained through $\nabla_{\mathbf{x}}\Xi(\mathbf{x}, \mathbf{y}^*) = 0$. Using (5.12) and (5.13), the dual function is obtained, which is given by (5.14), shown at the next page.

$$\begin{aligned}\Upsilon(\boldsymbol{\delta}^*, \boldsymbol{\beta}^*, \boldsymbol{\tau}^*, \boldsymbol{\sigma}^*) = -\sum_{n=1}^{N}\sum_{a=1}^{A} \frac{\Phi^2}{4\tau_{n,a}^*} - \frac{\sum_{n=1}^{N}\sum_{a=1}^{A} \frac{\mathcal{R}_{n,a}\Phi}{2\tau_{n,a}^*}}{\sum_{n=1}^{N}\sum_{a=1}^{A} \frac{\mathcal{C}_{n,a}\Phi}{2\tau_{n,a}^*}} \\ - \sum_{n=1}^{N} \delta_n^* m_n - \sum_{k=1}^{K} \beta_k^* - \sum_{n=1}^{N} \sigma_n^*.\end{aligned} \tag{5.14}$$

The dual function is a concave function on $\mathcal{X}_p^\sharp$. The canonical dual problem associated with (5.5) can be formulated as

$$\min\left\{\mathcal{P}(\mathbf{x}) | \mathcal{X}_p\right\} = \max\left\{\Upsilon(\boldsymbol{\delta}^*, \boldsymbol{\beta}^*, \boldsymbol{\tau}^*, \boldsymbol{\sigma}^*) | \mathcal{X}_p^\sharp\right\}. \tag{5.15}$$

Theorem 2. *If* $\mathcal{P}(\tilde{\mathbf{x}}) = \Upsilon(\tilde{\mathbf{y}}^*)$ *where* $\tilde{\mathbf{x}}$ *denotes the KKT point of the primal problem and* $\tilde{\mathbf{y}}^* = (\tilde{\boldsymbol{\delta}}^*, \tilde{\boldsymbol{\beta}}^*, \tilde{\boldsymbol{\tau}}^*, \tilde{\boldsymbol{\sigma}}^*) \in \mathcal{X}_p^\sharp$ *denotes the KKT point of the dual function, there exists a perfect duality relationship between the primal problem in* (5.5) *and its canonical dual problem.*

Proof. The proof directly extends from [111]. ■

Theorem 2 shows that the BIP in (5.5) is converted into a continuous space canonical dual problem which is perfectly dual to it. Moreover, the KKT point of the dual problem provides the KKT point of the primal problem.

Theorem 3. *(global optimality conditions): If* $\tilde{\mathbf{y}}^* = (\tilde{\boldsymbol{\delta}}^*, \tilde{\boldsymbol{\beta}}^*, \tilde{\boldsymbol{\tau}}^*, \tilde{\boldsymbol{\sigma}}^*) \in \mathcal{X}_p^\sharp$*, then* $\tilde{\mathbf{x}}$ *is a global minimizer of* $\mathcal{P}(\mathbf{x})$ *over* $\mathcal{X}_p$ *and* $\tilde{\mathbf{y}}^*$ *is a global maximizer of* $\Upsilon(\tilde{\boldsymbol{\delta}}^*, \tilde{\boldsymbol{\beta}}^*, \tilde{\boldsymbol{\tau}}^*, \tilde{\boldsymbol{\sigma}}^*)$ *over* $\mathcal{X}_p^\sharp$*. Hence,* $\mathcal{P}(\tilde{\mathbf{x}}) = \min\{\mathcal{P}(\mathbf{x})|\mathcal{X}_p\} = \max\left\{\Upsilon(\boldsymbol{\delta}^*, \boldsymbol{\beta}^*, \boldsymbol{\tau}^*, \boldsymbol{\sigma}^*)|\mathcal{X}_p^\sharp\right\} = \Upsilon(\tilde{\boldsymbol{\delta}}^*, \tilde{\boldsymbol{\beta}}^*, \tilde{\boldsymbol{\tau}}^*, \tilde{\boldsymbol{\sigma}}^*)$.

Proof. The proof directly extends from [111]. ■

According to Theorem 3, if the given global optimality conditions are met, the solution of the canonical dual problem provides an optimal solution to the primal problem. Solving the KKT conditions associated with the dual function in (5.14) is necessary and sufficient for global optimality as the dual problem is a concave maximization problem over $\mathcal{X}_p^\sharp$.

The KKT conditions of the dual function in (5.14) are given by $(\partial\Upsilon/\partial\delta_n^*) = 0, (\partial\Upsilon/\partial\beta_k^*) = 0, (\partial\Upsilon/\partial\tau_{n,a}^*) = 0$ and $(\partial\Upsilon/\partial\sigma_n^*) = 0$, where the respective partial derivatives are given by (5.16)-(5.19).

$$\frac{\partial\Upsilon}{\partial\delta_n^*} = -\sum_{n=1}^{N}\sum_{a=1}^{A}\frac{s_{n,a}\Phi}{2\tau_{n,a}^*} - \frac{\sum_{n=1}^{N}\sum_{a=1}^{A}\frac{\mathcal{R}_{n,a}s_{n,a}}{2\tau_{n,a}^*}}{\sum_{n=1}^{N}\sum_{a=1}^{A}\frac{\mathcal{C}_{n,a}\Phi}{2\tau_{n,a}^*}} + \frac{\sum_{n=1}^{N}\sum_{a=1}^{A}\frac{\mathcal{C}_{n,a}s_{n,a}}{2\tau_{n,a}^*}\cdot\sum_{n=1}^{N}\sum_{a=1}^{A}\frac{\mathcal{R}_{n,a}\Phi}{2\tau_{n,a}^*}}{\left(\sum_{n=1}^{N}\sum_{a=1}^{A}\frac{\mathcal{C}_{n,a}\Phi}{2\tau_{n,a}^*}\right)^2} - \sum_{n=1}^{N} m_n, \tag{5.16}$$

$$\frac{\partial \Upsilon}{\partial \beta_k^*} = -\sum_{n=1}^{N}\sum_{a=1}^{A}\left(\frac{\sum_{k=1}^{K} Y_{k,a}^n}{2\tau_{n,a}^*}\Phi\right) - \frac{\sum_{n=1}^{N}\sum_{a=1}^{A}\frac{\mathcal{R}_{n,a}\sum_{k=1}^{K} Y_{k,a}^n}{2\tau_{n,a}^*}}{\sum_{n=1}^{N}\sum_{a=1}^{A}\frac{\mathcal{C}_{n,a}\Phi}{2\tau_{n,a}^*}} + \frac{\sum_{n=1}^{N}\sum_{a=1}^{A}\frac{\mathcal{C}_{n,a}\sum_{k=1}^{K} Y_{k,a}^n}{2\tau_{n,a}^*}\cdot\sum_{n=1}^{N}\sum_{a=1}^{A}\frac{\mathcal{R}_{n,a}\Phi}{2\tau_{n,a}^*}}{\left(\sum_{n=1}^{N}\sum_{a=1}^{A}\frac{\mathcal{C}_{n,a}\Phi}{2\tau_{n,a}^*}\right)^2} - K, \tag{5.17}$$

$$\frac{\partial \Upsilon}{\partial \tau_{n,a}^*} = \sum_{n=1}^{N}\sum_{a=1}^{A}\left[\left(\frac{\Phi}{2\tau_{n,a}^*}\right)^2 + \left(\frac{\Phi}{2\tau_{n,a}^*}\right)\right] - \frac{\sum_{n=1}^{N}\sum_{a=1}^{A}\left(\frac{-\mathcal{R}_{n,a}\Phi}{2{\tau_{n,a}^*}^2} - \frac{\mathcal{R}_{n,a}}{2\tau_{n,a}^*}\right)\cdot\sum_{n=1}^{N}\sum_{a=1}^{A}\frac{\mathcal{R}_{n,a}\Phi}{2\tau_{n,a}^*}}{\left(\sum_{n=1}^{N}\sum_{a=1}^{A}\frac{\mathcal{C}_{n,a}\Phi}{2\tau_{n,a}^*}\right)^2} + \frac{\sum_{n=1}^{N}\sum_{a=1}^{A}\left(\frac{-\mathcal{C}_{n,a}\Phi}{2{\tau_{n,a}^*}^2} - \frac{\mathcal{C}_{n,a}}{2\tau_{n,a}^*}\right)}{\sum_{n=1}^{N}\sum_{a=1}^{A}\frac{\mathcal{C}_{n,a}\Phi}{2\tau_{n,a}^*}}, \tag{5.18}$$

$$\frac{\partial \Upsilon}{\partial \sigma_n^*} = -\sum_{n=1}^{N}\sum_{a=1}^{A}\frac{\Phi}{2\tau_{n,a}^*} - \frac{\sum_{n=1}^{N}\sum_{a=1}^{A}\frac{\mathcal{R}_{n,a}}{2\tau_{n,a}^*}}{\sum_{n=1}^{N}\sum_{a=1}^{A}\frac{\mathcal{C}_{n,a}\Phi}{2\tau_{n,a}^*}} + \frac{\sum_{n=1}^{N}\sum_{a=1}^{A}\frac{\mathcal{C}_{n,a}}{2\tau_{n,a}^*}\cdot\sum_{n=1}^{N}\sum_{a=1}^{A}\frac{\mathcal{R}_{n,a}\Phi}{2\tau_{n,a}^*}}{\left(\sum_{n=1}^{N}\sum_{a=1}^{A}\frac{\mathcal{C}_{n,a}\Phi}{2\tau_{n,a}^*}\right)^2} - N. \tag{5.19}$$

5.5.2 Invasive Weed Optimization Algorithm

Traditional gradient-based algorithms exist in literature for solving the non-linear equations resulting from the KKT conditions associated with the dual function. However, they show many defects such as oscillatory behavior, sensitivity to choice of initial values and complexity associated with the differentiation of KKT conditions and calculation of step size.

An IWO [27, 112] algorithm is used for solving the complex non-linear equations associated with the KKT conditions [112]. Inspired by the invasive and

robust nature of weeds, IWO is an evolutionary optimization algorithm, which has been shown to perform better than traditional approaches in terms of convergence. It also has the desirable properties of dealing with non-differentiable and complex objective functions and does not show the aforementioned defects.

In summary, the key steps of IWO are as follows:

- *Initialization*, where seeds are randomly dispersed over the search space;
- *Reproduction*, where every seed grows to a flowering plant and produces seeds;
- *Spatial Dispersion*, where produced seeds are distributed based on a normal distribution with a mean of zero and standard deviation reducing from an initial value σ_{initial} to a final value σ_{finala} according to equation $\sigma_{\text{iter}} = [(iter_{\max} - iter)/iter_{\max}]^g(\sigma_{\text{initial}} - \sigma_{\text{final}}) + \sigma_{\text{final}}$, where g is the modulation index;
- *Competitive Exclusion*, where a competitive mechanism is implemented for eliminating undesirable plants. A detailed discussion on IWO is out of scope of this study. Interested reader is referred to [27, 115].

5.6 Simulation Results

A hierarchical in-network caching system consisting of 4 levels is considered. Without loss of generality, the maximum possible storage capacities of hierarchical caching levels 1, 2, 3 and 4 are set to 200, 400, 500 and 600 gigabytes, respectively. In order to analyze the effects of maximum possible storage capacity on the performance of the proposed approach, the cache size is extended in increments of 20% until the maximum storage capacity of the first, second, third and forth level caches reach 600, 1200, 1500 and 1800 gigabytes (typical storage capacities available today). In defining the cost and return functions,

it is assumed that caching in the lower levels of the in-network caching system is more costly and results in more transmission bandwidth saving benefit. The total number of popular videos is considered to be 4000 with 3 popular quality layers. As in [116, 117], it is assumed that the video popularity is Zipf-like with a parameter of 0.6 and the video file sizes follow a Pareto (0.25) distribution with a minimum size of 60 megabytes.

The KKT conditions are solved for each dual variable associated with the dual problem deploying IWO and the allocation vector $\mathbf{x}_k$ is computed using (5.13). A pseudo code for the cache provisioning algorithm is given as Algorithm 3. TABLE 5.1 provides a summary of the simulation parameters for IWO.

Fig. 5.2 compares the effect of using a logarithmic function with a linear function in identifying the optimal provisioned storage size under maximum possible capacity varying from 1.7 to 5.1 terabytes (20% increments). In both scenarios, an increase in the storage capacity increases the identified provisioned cache size. We note that when a maximum possible capacity of approximately 3.7 terabytes is reached, the in-network caching system possesses most of the popular videos worthy of being cached. Therefore, further increasing the maximum storage capacity does not lead to a noticeable increase in the provisioned cache size at this point.

The proposed approach is compared with the case when no storage provisioning is performed within the hierarchical in-network caching system and popular contents are cached using LFU caching algorithm [118]. LFU caches the most popular videos in the lower level caches closer to the end users [119]. In contrast with the other widely used caching algorithm, LRU, LFU focuses on historical popularity over a long period of time. As a cache provisioning technique, the proposed approach also considers a long term content popularity. Therefore, it is pertinent to this scheme with LFU.

Fig. 5.3 compares the performance of the proposed approach with LFU in

Algorithm 3: Hierarchical caching based on IWO (adapted from [112])

initialize $\boldsymbol{\delta}^*, \boldsymbol{\beta}^*, \boldsymbol{\tau}^*, \boldsymbol{\sigma}^*, \forall n \in \mathcal{N}, iter = 0$;

$\forall\ \partial\Upsilon/\partial\boldsymbol{\nu}^*$, where $\boldsymbol{\nu}^* \in (\boldsymbol{\delta}^*, \boldsymbol{\beta}^*, \boldsymbol{\tau}^*, \boldsymbol{\sigma}^*)$

create initial population of Q individuals (weeds): $\mathcal{W} = \{W_1, \cdots, W_Q\}$;

while $|\boldsymbol{\nu}^*| > \varrho$ *or* $iter = iter_{max}$ **do**

evaluate the fitness of each individual i.e., calculate $f(W_n), \forall n \in \mathcal{W}$;

sort $\mathcal{W}$ in ascending order according to $f(W_n)$;

select the first Q_p individuals of $\mathcal{W}$ to create the set $\mathcal{W}_p$;

$\forall\ W_j, j = 1, \cdots, Q_p$

generate

$S_j = \frac{f(W_j) - f_{\text{worst}}}{f_{\text{best}} - f_{\text{worst}}} \times (S_{\max} - S_{\min}) + S_{\max}$ seeds;

create population of generated seeds, $\mathcal{W}_s = \{W_s\}$;

for $i = 1 : |\mathcal{W}_s|$ **do**

$W_s^i \leftarrow W_s^i + \phi^i$, where $\phi^i \sim L(0, \sigma_{\text{iter}})$;

end

create $\mathcal{W}^* = \mathcal{W} \cup \mathcal{W}_s$;

sort $\mathcal{W}^*$ in ascending order according to fitness;

select the first $Q_{\max}$ individuals of $\mathcal{W}^*$ and create $\mathcal{W}$;

end

select the best fitted individuals $\boldsymbol{\delta}^*, \boldsymbol{\beta}^*, \boldsymbol{\tau}^*$ and $\boldsymbol{\sigma}^*$; calculate $\mathbf{x}_n$ using (5.13);

TABLE 5.1: IWO Numerical Parameter Values

Parameter	Value
Size of initial population (Q)	20
Min. fitness threshold (ϱ)	10^{-7}
Maximum number of iterations ($iter_{\max}$)	500
Maximum number of plants ($Q_{\max}$)	10
Minimum number of seeds ($S_{\min}$)	0
Maximum number of seeds ($S_{\max}$)	5
Non-linear modulation index	2.5
Initial standard deviation (σ_{initial})	10
Final standard deviation (σ_{final})	0.01

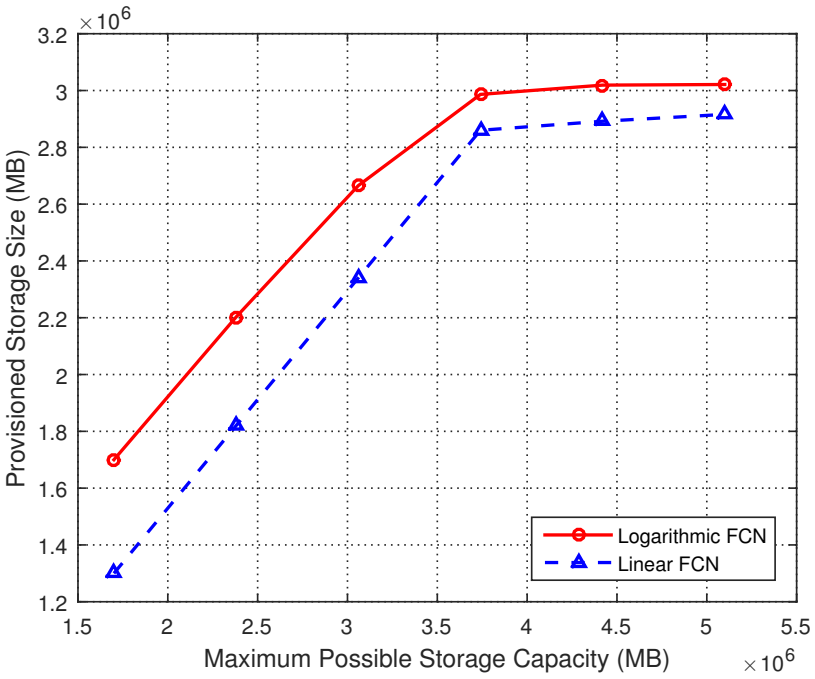


Fig. 5.2: Provisioned storage vs. maximum possible storage.

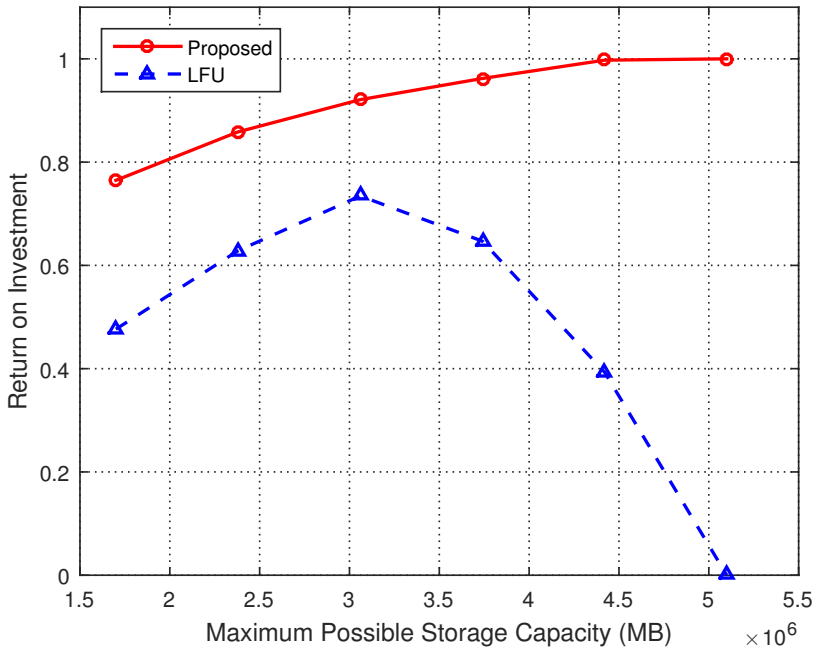


Fig. 5.3: Return on investment vs. maximum possible storage.

terms of return on investment under different maximum possible storage capacities mentioned earlier. It is noted that the proposed approach improves return to investment ratio by 43.74%. When there is approximately 3.1 terabytes of storage capacity available, the return on investment performance of LFU starts degrading as by this point, most of the popular videos have been cached and adding more storage only increases the cost for the same amount of saving in transmission bandwidth.

Fig. 5.4 compares the storage cost-effectiveness of the proposed approach with LFU. Since LFU does not support intelligent storage provisioning and uses the maximum storage capacity available, extending the cache size in increments of 20% increases the storage cost exponentially. However, the proposed scheme only uses an optimal portion of the maximum possible storage and hence, decreases the cost significantly. The proposed approach improves cost-effectiveness by 38.59%. When there is 3.7 terabytes of storage available, the cost starts decreasing in the proposed approach as there is more storage available on cheaper caches at

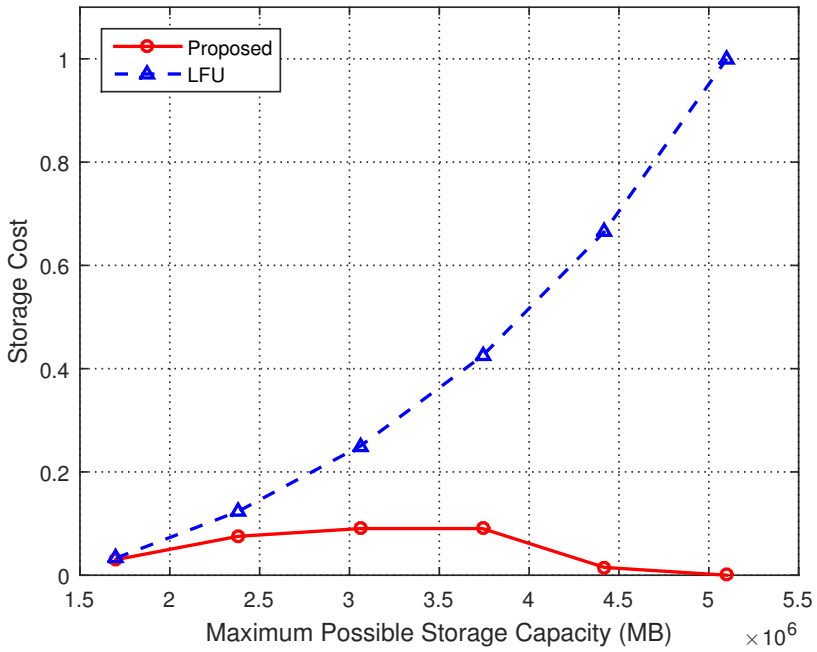


Fig. 5.4: Storage cost vs. maximum possible storage.

higher levels. Therefore, to increase cost-effectiveness, some of the videos that were previously cached at the expensive lower level caches move to the higher levels.

Fig. 5.5 indicates how the increase in the maximum storage capacity affects the provisioned storage size of the caches at each hierarchical level of the in-network caching system. As more storage is available on the cheaper devices in higher levels, more provisioned storage size is allocated to the higher level devices due to greater cost efficiency.

Fig. 5.6 compares the reduction in inter and intra-ISP traffic as a result of deploying the proposed approach and LFU caching mechanism. It can be seen that LFU performs slightly better in terms of load reduction by only 0.764%, at the cost of considerably higher available storage, resulting in a significant increase in cost. It is worth noting that with LFU, upon availability of approximately 3.1 terabytes storage, most of the popular videos are cached and an increase in the maximum storage capacity does not further reduce the load in the CN.

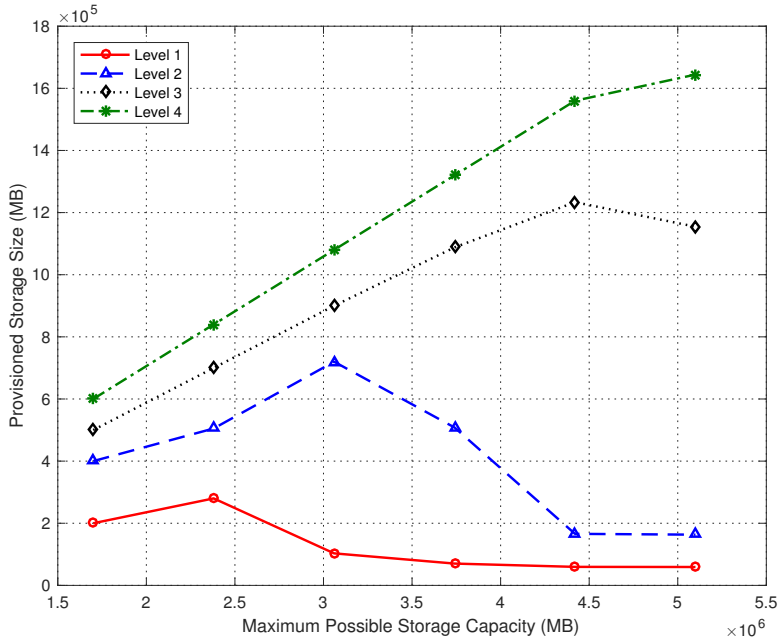


Fig. 5.5: Provisioned storage of different levels of hierarchical caching system vs. maximum possible storage.

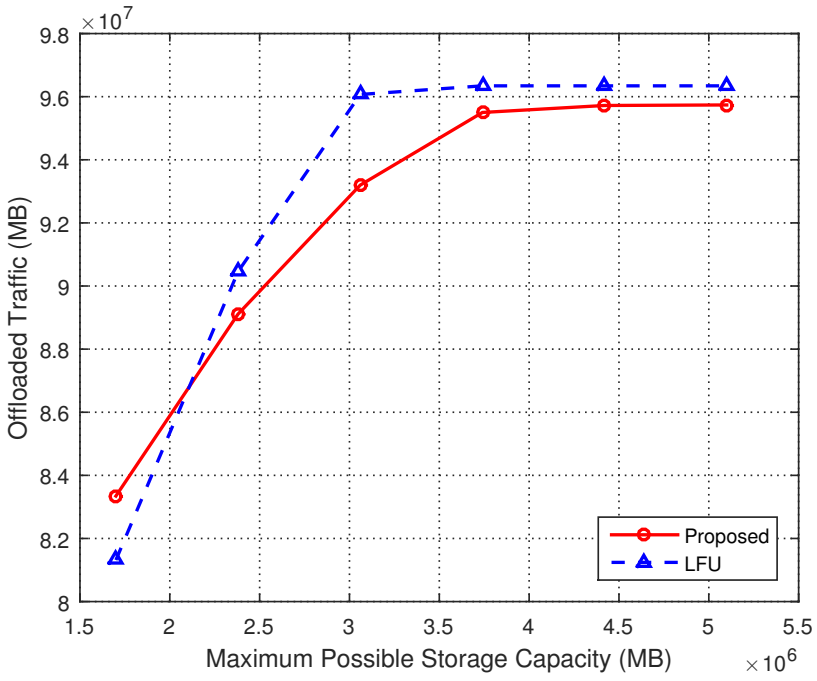


Fig. 5.6: Inter and intra-ISP traffic reduction vs. maximum possible storage.

5.6.1 Complexity Analysis

IWO is an iterative algorithm and is used for each dual variable associated with the dual function in (5.14). In each iteration for $\boldsymbol{\delta}^* \geq 0, \boldsymbol{\beta}^* \geq 0, \boldsymbol{\tau}^* \geq 0, \boldsymbol{\sigma}^* \geq 0$, N, K, NA, and N variables are computed, respectively. Therefore, it has an overall worst case complexity of $\mathcal{O}(iter_{\max} \cdot \{2N + K + NA\})$ [112].

[27] and [115] conduct a comprehensive assessment of the performance of IWO algorithm in terms of convergence and computational time through comparison with Genetic Algorithm, Particle Swarm Optimization, Differential Evolution and other evolutionary algorithms.

5.7 Conclusion

This chapter has proposed a cost-effective cache provisioning scheme, which optimizes cache storage allocation inside a hierarchical in-network caching system, in order to minimize both storage and transmission bandwidth costs. CDT is used to convert our BIP problem into its canonical dual. The IWO algorithm is deployed to obtain the solution of the dual problem. Numerical and simulation results have shown that the proposed scheme outperforms LFU algorithm by more than 43% and 38% in terms of return on investment and cost-efficiency improvement, respectively.

The recent trend of virtualizing mobile network functions into software-based cloud servers motivates the research on CaaS, which is discussed in detail in the next chapter.

Chapter 6

Cost-Driven Mobile Video Caching-as-a-Service

6.1 Introduction

In the previous chapter, a cache provisioning problem, which finds the best trade-off between the cost of cache storage and bandwidth savings from hierarchical caching was formulated. Recently, a new trend of virtualizing mobile network functions into software-based cloud servers, has emerged. For instance, with EPCaaS, some EPC network functions are instantiated on VMs on top of a virtualized platform, running in an operator's cloud center [19]. The increasing drive towards mobile network function virtualization has also motivated the CaaS research, which offers content caching capabilities inside the MNOs' cloud centers [20]. In contrast to traditional CDNs, CaaS approach offers various levels of flexibility for service providers (SPs) and CPs. It also has several advantages over traditional in-network caches such as optimization of resource utilization, reduction in capital expenditures (CAPEX) and operating expenditures (OPEX), in addition to an increase in scalability and flexibility [19–22]. CaaS instances in mobile cloud centers can be adaptively created, migrated, scaled (up or down),

shared and released on-demand. In this model, MNO may charge CPs and SPs for caching the content in a mobile cloud based on some service-level agreement (SLA) [120].

In this chapter, a virtual caching policy for cloud-based mobile operator networks, which maximizes the return on caching investments is proposed. The return function is formulated based on the reduced traffic volumes, which in the absence of caching mechanisms in the operator's network have to be served by CDNs or content providers directly, therefore, inducing corresponding content distribution costs. This is the first cost-driven CaaS approach that has been proposed for cloud-based mobile networks in proactive off-line scenarios, i.e. when caching decisions are made in advance based on the expected popularity of content items [120].

6.2 Contributions and Outline

The main contributions of this chapter can be summarized as follows:

- a virtual caching optimization framework, namely maximum return on investment (MRI), is formulate, which maximizes the return on caching investment. The proposed budget-constrained approach (maximum offloaded traffic (MOT)) maximizes the offloaded traffic, while meeting the maximum budget threshold. More specifically, taking the popularity and size of video contents into account, MRI and MOT aim to find the optimal caching tables which would maximize the ratio of transmission bandwidth cost to storage cost and the offloaded traffic for a given budget, respectively.

- by introducing a video quality weighting factor in the optimization problem, the key QoE differentiators (e.g. higher throughput, lower latency, smaller start up and buffering times [28, 29]) in delivering content items to the end users are taken into account.

- this chapter also focuses on SVC-based DASH video encoding, which encodes a single video into different quality layers and, thus, provides a more resource-efficient alternative to the traditional H.264/AVC-based DASH encoding in which a separate AVC video file is encoded for every video quality format [2].

- the virtual caching problem is solved using CDT [26]. More specifically, the proposed BIP problem is transformed into a canonical dual problem in continuous space, which is a concave problem. Additionally, the conditions under which the solutions of the canonical dual problem and primal problem are identical are provided.

- the canonical dual problem results in complex non-linear equations, which are efficiently solved by applying the IWO algorithm [27].

The rest of the chapter is structured as follows. The related work is summarized in Section 6.3. Section 6.4 describes the system model. The virtual caching framework is formulated in 6.5. Section 6.6 presents the canonical dual framework. Section 6.7 conducts a simulation analysis of the model. This chapter finishes by a conclusion in Section 6.8.

6.3 Related Work

Many studies have proposed CDNs for Internet content [121,122], as well as CDN services running in the cloud [123,124]. However, as explained earlier, caching in Internet CDNs does not address the problems of latency and capacity for video delivery in wireless networks.

Some research has been carried out on caching web content in cellular networks [125] and on mobile devices [126]. However, in these studies, the challenges of video delivery and caching at the network's edge have not been taken into account.

Krishnappa et al. in [127] investigate the effectiveness of video caching using LFU, LRU and a combination of the two deploying traces of Hulu. Nevertheless, like the Internet caching techniques, the aforementioned schemes do not address the problem of delay or video capacity in mobile networks.

Some studies have developed caching techniques for ad hoc networks [126,128]. However, the applicability of these techniques to the problem of video caching and delivery in mobile networks is questionable.

Several approaches have been proposed to analyze intelligent caching strategies for mobile content caching inside MNO's network [12, 14]. An extensive overview of the techniques for in-network content caching in 5G mobile networks has been introduced in [15], whereas different proactive mobile caching schemes in BSs have been discussed in [10, 13, 129–131]. These works however, do not address the problem of caching in a cloud-based mobile network. Furthermore, these theoretical studies for in-network caching and caching content in BSs lack practical implementation consideration. For instance, the caching approach proposed in [129] needs the presence of additional helper nodes where videos are cached, and for users to have access to multiple helper nodes, both of which may be hard to satisfy.

Reference [20] represents the first attempt to develop a virtualized caching system inside MNOs' cloud center. The differences between this study and the work of [20] are fourfold: 1) the work in [20] only minimizes inter- and intra-MNO traffic load and does not take cost-efficiency and caching costs into account; 2) reference [20] does not take the SVC video requirements into consideration; 3) the constraints on the capacity of the fronthaul are not taken into account in [20]; 4) the virtual caching problem proposed in [20] is solved using a simplistic algorithm, which runs relatively fast, however, rarely achieves an optimal allocation [132];

6.4 System Model

This chapter considers a cloud-based virtual caching system inside the MNO's infrastructure which operates as follows (Fig. 6.1). If a content item is not available in the MNO's virtual cache, it needs to traverse the MNO's core and virtual BBU pool to get to the RRHs in a cluster, from which it is transmitted to the end users. Likewise, in order to cache a content in the operator's network, it needs to travel through the MNO core to be cached in the BBU pool, from which it is sent to the RRHs to be transmitted to the end users. The requests for the content are then served from the BBU pool in the MNO's infrastructure. The third-party SPs and CPs can program the virtual caching using CaaS's application programming interfaces (APIs). A SLA is defined between the MNO and CPs, which determines the MNO's liabilities in providing the required resources to guarantee a level of service for the videos that have been cached. It is also assumed that the MNO can dynamically charge for the resource utilization of the SPs and CPs.

The system consists of I video streams, which are indexed by the set $\mathcal{I} \triangleq \{1, \ldots, i, \ldots, I\}$. Quality layers of a video stream are indexed by the set $\mathcal{J} \triangleq \{1, \ldots, j, \ldots, J\}$. $q_{i,j}$ denotes the j^{th} quality layer of video object i, which has a size, source bit rate and popularity (hit rate) of $f_{i,j}$, $b_{i,j}$ and $p_{i,j}$, respectively. Different clusters are indexed by $\mathcal{N} \triangleq \{1, \ldots, n, \ldots, N\}$. One example of a cloud-based caching system architecture can be found in [20].

6.4.1 Notations and Variables

Cache Assignment Binary Decision Variable ($x_{n,i,j}$)

$x_{n,i,j}$ represents an entry in the caching table $\mathbf{x}$. $x_{n,i,j} = 1$ indicates that content $q_{i,j}$ is cached to serve users in cell n while meeting the SLA on users' experience of the content. If $x_{n,i,j} = 0$ but content $q_{i,j}$ is available in the cache

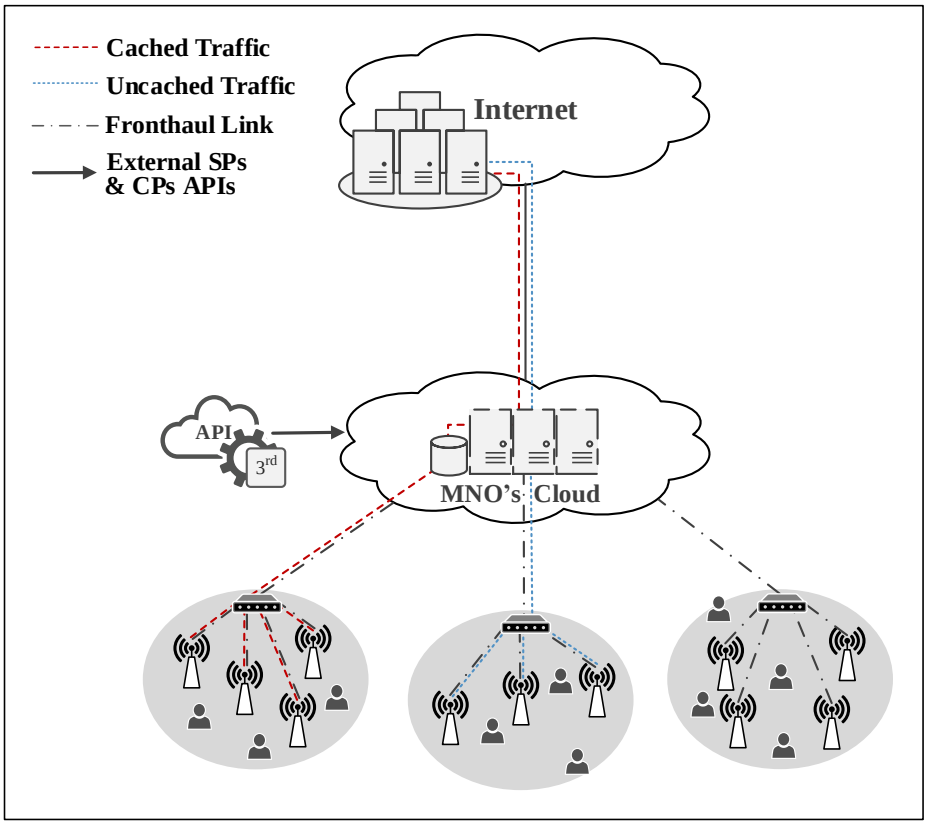


Fig. 6.1: Cloud-based virtual caching architecture.

($\sum_{n=1}^{N} x_{n,i,j} \geq 1$) to serve users in a cell n' under SLA guarantees, requests for content $q_{i,j}$ from users in cell n can be directed to the cache without any SLA liabilities. If $\sum_{n=1}^{N} x_{n,i,j} = 0$, requests for content $q_{i,j}$ are routed to the root.

Offloaded Traffic ($l_{i,j}$)

$l_{i,j}$ is the traffic load that would be directed to public CDNs in the absence of virtual caching in the MNO's network. $l_{i,j}$ is the reduction in the transmission bandwidth as a result of caching $q_{i,j}$, where $l_{i,j} = f_{i,j} \cdot p_{i,j}$. $\mathtt{L}_n$ denotes the cached traffic for each cluster n, which is given by $\mathtt{L}_n = \sum_{i=1}^{I} \sum_{j=1}^{J} l_{i,j} \cdot x_{n,i,j} \quad \forall\, n \in \mathcal{N}$.

Storage Size ($\mathtt{S}_n$)

$\mathtt{S}_n$ is the storage capacity allocated to cluster n for cloud-based caching. For pricing purposes, the required storage is calculated under the assumption that cached files are not shared between different clusters. The total storage required for an in-

dividual cluster n is $\mathtt{S}_n = \sum_{i=1}^{I} \sum_{j=1}^{J} f_{i,j} \cdot x_{n,i,j} \; \forall n \in \mathcal{N}$. The binary decision variable $y_{i,j}$ is used to find the total physical storage required $\mathcal{S} = \sum_{i=1}^{I} \sum_{j=1}^{J} f_{i,j} \cdot y_{i,j}$, where $y_{i,j} \; \forall i, j$ is given by

$$y_{i,j} = \begin{cases} 1 & \text{if } \sum_{i=n}^{N} x_{n,i,j} \geq 1 \\ 0 & \text{otherwise.} \end{cases} \tag{6.1}$$

Fronthaul Capacity ($B_n^{\max}$)

$B_n^{\max}$ is the bandwidth capacity of the link between the operator's cloud center and cluster n. It should be noted that in order to meet the SLA with CPs, the MNO needs to provision for the peak rather than average bandwidth.

Quality Priority Factor ($\mathcal{Q}$)

$\mathcal{Q}$ prioritizes the video contents with higher bit rates over low bit rate videos. The offered throughput under TCP is inversely proportional to connection's round trip time (RTT) [133]. As shown in [13], in comparison with fetching data from public CDNs, caching contents inside the MNO's infrastructure results in a considerable decrease in RTT. Therefore, in order to allocate higher bandwidth to video contents with high bit rate requirements, high bit rate contents are cached closer to the end users, which increases their TCP throughput and consequently reduces latency. The quality priority weighting factor estimates the summation of the bit rates of cached contents normalized over sum of bit rates of all video contents as follows:

$$\mathcal{Q} = \frac{\sum_{n=1}^{N} \sum_{i=1}^{I} \sum_{j=1}^{J} b_{i,j} \cdot x_{n,i,j}}{N \cdot \sum_{i=1}^{I} \sum_{j=1}^{j} b_{i,j}}. \tag{6.2}$$

Return Function ($\mathcal{R}$)

$\mathcal{R}$ is the benefit gained from the virtualized caching system, which lies in the fact that caching video contents in the MNO's infrastructure would minimize the traffic load that would be directed to public CDNs. As customers of these CDNs, CPs are charged on the basis of the amount of traffic that is served from the CDN. It is assumed that the benefit of transmission bandwidth saving follows a predefined function $\Gamma : \mathbb{R} \rightarrow \mathbb{R}$. Thus, the benefit derived from the reduction in transmission bandwidth (hereinafter offloaded traffic) is estimated when videos are cached for cluster n of the virtual caching system as

$$\mathcal{R}\left(\mathtt{L}_n\right) = \Gamma\left(\sum_{i=1}^{I}\sum_{j=1}^{J} l_{i,j} \cdot x_{n,i,j}\right) \qquad \forall n \in \mathcal{N}. \tag{6.3}$$

Cost Function ($\mathcal{C}$)

$\mathcal{C}$ is the cost incurred, which lays in the amount of storage that is required for caching video contents. In general, public CDNs charge their customers based on the amount of bandwidth served by them. However, as the traffic load would traverse the MNO's infrastructure whether or not the contents are cached, the main factor incurring cost would be the cost of storage. It is assumed that the cache storage cost follows a predefined function $\Lambda : \mathbb{R} \rightarrow \mathbb{R}$. Hence, the cost associated with provisioned storage size $\mathtt{S}_n$ is given by

$$\mathcal{C}\left(\mathtt{S}_n\right) = \Lambda\left(\sum_{i=1}^{I}\sum_{j=1}^{J} f_{i,j} \cdot x_{n,i,j}\right) \qquad \forall n \in \mathcal{N}. \tag{6.4}$$

Both benefit and cost functions can be any appropriate function defined by the operator.

A summary of commonly used notation is provided in TABLE 6.1.

TABLE 6.1: Commonly Used Notation

Notation	Description
i	Video object index
j	Quality layer index of a video object
n	Cluster index
I	Total number of video objects
J	Total number of quality layers of a video object
N	Total number of clusters in the network
$q_{i,j}$	The j^{the} quality layer of video content i
$x_{n,i,j}$	A binary decision variable indicating whether video content q_{ij} is cached for cluster n
$f_{i,j}$	Size of the j^{th} quality layer of video object i
$p_{i,j}$	Popularity of the j^{th} quality layer of video object i
$b_{i,j}$	Source bit rate of the j^{th} quality layer of video i
$l_{i,j}$	Offloaded traffic of cluster n
$\mathcal{Q}$	Quality priority weighting factor
$\mathtt{L}_n$	Sum offloaded traffic of cluster n
$\mathtt{S}_n$	Size of cache storage of cluster n
$\mathcal{R}$	Offloaded traffic return function
$\mathcal{C}$	Cache storage cost function
$C^{\max}$	Total caching budget
$B_n^{\max}$	Link capacity of fiber line to RRH n

6.5 Problem Formulation

In this section, two virtual proactive caching problems are formulated based on the system model introduced in Section 6.4. The first optimization problem is formulated to achieve the optimal trade-off between the cost of caching video contents (investment) and the benefit gained from content caching (return). The second optimization problem aims to maximize the offloaded traffic under the constraint of the total caching budget.

6.5.1 Return on Investment Maximized Caching

The caching problem aims to maximize the return on investment [hereinafter referred to as maximum return on investment (MRI)] as follows:

$$\max_{\mathbf{x}} \ \mathcal{Q} \cdot \frac{\sum_{n=1}^{N} \mathcal{R}\left(\mathtt{L}_n\right)}{\sum_{n=1}^{N} \mathcal{C}\left(\mathtt{S}_n\right)} \tag{6.5}$$

subject to:

$$\sum_{i=1}^{I}\sum_{j=1}^{J} b_{i,j} \cdot p_{i,j} \cdot x_{n,i,j} \leq B_n^{\max} \qquad \forall n \in \mathcal{N} \tag{6.5a}$$

$$x_{nij-1} \geq x_{n,i,j} \qquad \forall n \in \mathcal{N}, \forall i \in \mathcal{I}, \forall j \in \mathcal{J}_{-\{1\}} \tag{6.5b}$$

$$x_{n,i,j} \in \{0,1\} \qquad \forall n \in \mathcal{N}, \forall i \in \mathcal{I}, \forall j \in \mathcal{J}. \tag{6.5c}$$

The objective of optimization problem (6.5) is to find the optimal caching table $\mathbf{x}$ which determines what content should be cached for which cluster in order that the ratio of overall return (6.3) to overall cost (6.4) is maximized. Constraint (6.5a) ensures that sum of bit rates of the video objects cached in the caching system for cluster n is upper-bounded by the maximum fronthaul capacity threshold, $B_n^{\max}$. This ensures adequate provision for peak bandwidth. Constraint (6.5b) ensures that if a video quality layer is cached, all the lower quality layers are cached too (SVC requirement). Binary variables $x_{n,i,j} \in \{0,1\}$

is used, as explained in Section 6.4.1.

6.5.2 Budget-Constrained Caching

The budget-constrained caching problem, namely maximum offloaded traffic (MOT) is formulated as follows:

$$\max_{\mathbf{x}} \sum_{n=1}^{N}\sum_{i=1}^{I}\sum_{j=1}^{J} l_{i,j} \cdot x_{n,i,j} \tag{6.6}$$

subject to:

$$\sum_{n=1}^{N} \mathcal{C}(\mathtt{S}_n) \leq C^{\max} \tag{6.6a}$$

$$\sum_{i=1}^{I}\sum_{j=1}^{J} b_{i,j} \cdot p_{i,j} \cdot x_{n,i,j} \leq B_n^{\max} \qquad \forall n \in \mathcal{N} \tag{6.6b}$$

$$x_{nij-1} \geq x_{n,i,j} \qquad \forall n \in \mathcal{N}, \forall i \in \mathcal{I}, \forall j \in \mathcal{J}_{-\{1\}} \tag{6.6c}$$

$$x_{n,i,j} \in \{0,1\} \qquad \forall n \in \mathcal{N}, \forall i \in \mathcal{I}, \forall j \in \mathcal{J}. \tag{6.6d}$$

The objective of optimization problem (6.6) is to find the optimal caching table $\mathbf{x}$ which maximizes the amount of cached traffic. (6.6a) represent the budget constraint. Constraints (6.6b)-(6.6d) are identical to the constraints in (6.5).

Herein, optimization problem (6.5) is solved. Problem (6.6) can be solved similarly. However, due to limitations in space, the solution to (6.6) is omitted. Problem (6.5) is difficult to solve due to its combinatorial nature. As an intermediate step towards solution, by defining a cache allocation matrix, (6.5) is converted into a BIP problem, where instead of making decisions on the basis of individual video quality layer, decisions are made on the basis of feasible set of video layer cache allocation patterns that satisfies constraint (6.6c). The idea of pattern allocation is similar to [110, 134]. All the combinations of video streams and the respective quality layers are indexed by the set $\mathcal{K} \triangleq \{1, \ldots, k, \ldots, K\}$. $K = |\mathcal{K}|$ denotes the cardinality of set $\mathcal{K}$. The cache allocation matrix is of

the order $K \times A$, where each row corresponds to the video stream-video quality layer combination index and each column corresponds to a feasible cache allocation pattern [meeting constrain (6.6c)]. A denotes the total number of feasible allocation patterns. The basic idea of this cache allocation matrix, for the case of 2 videos each with 2 quality layers is illustrated by (6.7). In any allocation pattern (i.e., any column), a "1" is placed when the video quality layer is cached, otherwise a "0" is placed.

$$\mathbf{Y}^n = \begin{bmatrix} 1 & 1 & 0 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 0 & 1 \end{bmatrix} \tag{6.7}$$

A cache indicator vector $\mathbf{x} \triangleq [\mathbf{x}_n]_{N\times 1}$ is defined, where $\mathbf{x}_n = [x_{n,a}]_{A\times 1}$. Each entry $x_{n,a} \in \{0, 1\}$ indicates whether the cache allocation pattern a is allocated for cluster n or not.

Note that all the clusters in the virtual caching system have the same cache allocation patterns matrix. (6.5) is rewritten as a BIP problem as follows:

$$\min_{\mathbf{x}} \left\{ \mathcal{P}(\mathbf{x}) = -\mathcal{Q} \cdot \frac{\sum_{n=1}^{N} \sum_{a=1}^{A} \mathcal{R}(\mathtt{L}_{n,a}) \cdot x_{n,a}}{\sum_{n=1}^{N} \sum_{a=1}^{A} \mathcal{C}(\mathtt{S}_{n,a}) \cdot x_{n,a}} \right\} \tag{6.8}$$

subject to:

$$\sum_{a=1}^{A} b_{na} \cdot x_{n,a} \leq B_n^{\max} \qquad \forall n \in \mathcal{N} \tag{6.8a}$$

$$x_{n,a} \cdot (x_{n,a} - 1) = 0 \qquad \forall n \in \mathcal{N}, \forall a \tag{6.8b}$$

$$\sum_{a=1}^{A} x_{n,a} = 1 \qquad \forall n \in \mathcal{N}, \tag{6.8c}$$

where $\mathcal{Q} = \sum_{n=1}^{N} \sum_{a=1}^{A} (\mathcal{R}_{n,a} \cdot x_{n,a}/B)$, $\mathcal{R}_{n,a}$ and $\mathcal{C}_{n,a}$ are the transmission band-

width benefit and storage cost of allocating pattern a to cluster n. For a cluster n, (6.8a) puts an upper-bound of $B_n^{\max}$ on the fronthaul bandwidth capacity, which is equivalent to (6.5a). Constraint (6.8b) is a pure binary constraint that ensures $x_{n,a} \in \{0, 1\}$. Constraint (6.8c) ensures that at most one allocation pattern is chosen for each caching level.

Although the optimization problem (6.8) is simpler and more tractable than (6.5), the solution is still exponentially complex.

6.6 Canonical Dual Framework

6.6.1 Dual Problem Formulation

The BIP problem (6.8) is converted into a continuous space canonical dual problem using CDT [26, 111], which is solved in continuous space. The conditions under which the solution of the canonical dual problem is identical to that of the primal is then identified.

The feasible space for primal problem (6.8) is defined by $\mathcal{Z}_p = \left\{\mathbf{x} \in \{0, 1\}^{NA}\right\}$. The equality constraints (6.8b) and (6.8c) are temporarily relaxed to inequalities and the primal problem is transformed with these inequality constraints into continuous domain canonical dual problem. The problem is then solved in continuous space and the conditions under which the solutions of the canonical dual problem and primal problem are identical are provided.

As a key step towards canonical dual formulation, the geometrical operator for the primal problem is defined as $\wedge(\mathbf{y}) = (\boldsymbol{\delta}, \boldsymbol{\beta}, \boldsymbol{\tau}, \boldsymbol{\sigma}) \in \mathcal{Y}_g$, which is a vector

valued mapping where $\mathcal{Y}_g$ is the feasible space for $\mathbf{y}$, and

$$\begin{cases} \boldsymbol{\delta} = [\sum_{a=1}^{A} \mathcal{R}_{n,a} x_{n,a} - B_n^{\max}]_{N\times 1} \\ \boldsymbol{\beta} = [x_{n,a} \cdot (x_{n,a} - 1)]_{NA\times 1} \\ \boldsymbol{\tau} = [\sum_{a=1}^{A} x_{n,a} - 1]_{N\times 1} \end{cases} \tag{6.9}$$

Therefore, the feasible space for $\mathbf{y}$ is defined by $\mathcal{Y}_g = \mathbb{R}^N \times \mathbb{R} \times \mathbb{R}^{NA} \times R^N | \boldsymbol{\delta} \leq 0, \boldsymbol{\beta} \leq 0, \boldsymbol{\tau} \leq 0$. $\boldsymbol{\beta} = \sum_{n=1}^{N} \sum_{a=1}^{A} \mathcal{C}(\mathtt{S}_{n,a}) \cdot x_{n,a} - C^{\max}$.

Next, the indicator function [113] is defined as

$$V(\mathbf{y}) = \begin{cases} 0 & \text{if } \mathbf{y} \leq 0 \\ +\infty & \text{otherwise.} \end{cases} \tag{6.10}$$

The primal problem (6.8) is rewritten in the canonical form using indicator function (6.10) as follows:

$$\min \left\{ V(\wedge(\mathbf{y})) + \mathcal{P}(\mathbf{x}) \right\}. \tag{6.11}$$

$\mathbf{y}^* = (\boldsymbol{\delta}^*, \boldsymbol{\beta}^*, \boldsymbol{\tau}^*)$ is now defined as the vector of dual variables associated with the corresponding restrictions $\mathbf{y} \leq 0$. The feasible space for $\mathbf{y}^*$ is defined by $\mathcal{Y}_d = \mathbb{R}^N \times \mathbb{R}^{NA} \times R^N | \boldsymbol{\delta}^* \geq 0, \boldsymbol{\beta}^* \geq 0, \boldsymbol{\tau}^* \geq 0$. Based on the Fechnel transformation, the canonical sup-conjugate function of $V(\mathbf{y})$ is defined as

$$\begin{aligned} V^*(\mathbf{y}^*) &= \sup \left\{ \langle \mathbf{y}, \mathbf{y}^* \rangle - V(\mathbf{y}) | \mathbf{y} \in \mathcal{Y}_g, \mathbf{y}^* \in \mathcal{Y}_d \right\} \\ &= \sup_{\mathbf{y}^*} \left\{ \langle \boldsymbol{\delta}^T \boldsymbol{\delta}^* + \boldsymbol{\beta}^T \boldsymbol{\beta}^* + \boldsymbol{\tau}^T \boldsymbol{\tau}^* - V(\mathbf{y}) \rangle \right\} \\ &= \begin{cases} 0 & \text{if } \boldsymbol{\delta}^*, \boldsymbol{\beta}^*, \boldsymbol{\tau}^* \\ +\infty & \text{otherwise.} \end{cases} \end{aligned} \tag{6.12}$$

Using the definition of sub-differential, it can be easily verified that if $\mathbf{y}^* > 0$,

then the condition $\mathbf{y}^T\mathbf{y}^* = 0$ leads to $\mathbf{y} = 0$, and consequently $\mathbf{x} \in \mathcal{X}_p$. Hence, the dual feasible space for the primal problem in (6.8) is an open positive cone defined by $\mathcal{X}_p^\sharp = \{\mathbf{y}^* \in \mathcal{Y}_d | \mathbf{y}^* > 0\}$.

The total complementarity function [26] is defined as

$$\Xi(\mathbf{x}, \mathbf{y}^*) = \wedge(\mathbf{x})^T\mathbf{y}^* - V^*(\mathbf{y}^*) + \mathcal{P}(\mathbf{x}), \tag{6.13}$$

which is obtained by replacing $V(\mathbf{y}) = \wedge(\mathbf{x})^T\mathbf{y}^* - V^*(\mathbf{y}^*)$ (Fechnel-Young equality) in (6.11). The definitions of $\wedge(\mathbf{x}), V^*(\mathbf{y}^*)$ and $\mathcal{P}(\mathbf{x})$ are used to express $\Xi(\mathbf{x}, \mathbf{y}^*) = \Xi(\mathbf{x}, \boldsymbol{\delta}^*, \boldsymbol{\beta}^*, \boldsymbol{\tau}^*)$ as given by (6.14).

$$\begin{aligned}\Xi(\mathbf{x}, \mathbf{y}^*) = &\sum_{n=1}^{N}\sum_{a=1}^{A}\left[x_{n,a}\left(\delta_n^*\mathcal{R}_{n,a} + \tau_{n,a}^* - \beta_k^*\right)\right] \\ &- \frac{\sum_{n=1}^{N}\sum_{a=1}^{A}\mathcal{R}_{n,a}x_{n,a} \cdot \sum_{n=1}^{N}\sum_{a=1}^{A}\mathcal{R}(\mathtt{L}_{n,a})x_{n,a}}{B \cdot \sum_{n=1}^{N}\sum_{a=1}^{A}\mathcal{C}(\mathtt{S}_{n,a})x_{n,a}} \\ &- \sum_{n=1}^{N}\delta_n^*B_n^{\max} - \sum_{n=1}^{N}\tau_{n,a}^* + \sum_{n=1}^{N}\sum_{a=1}^{A}\beta_k^*x_{n,a}^2.\end{aligned} \tag{6.14}$$

Next, the canonical dual function [26, 113] is defined using the canonical dual variables as

$$\Upsilon(\boldsymbol{\delta}^*, \boldsymbol{\beta}^*, \boldsymbol{\tau}^*) = \mathrm{sta}\left\{\Xi(\mathbf{x}, \boldsymbol{\delta}^*, \boldsymbol{\beta}^*, \boldsymbol{\tau}^*)\right\}, \tag{6.15}$$

where $\mathrm{sta}(\cdot)$ denotes finding the stationary point of the function. We are primarily interested in the cache allocation vector $\mathbf{x}$ for a node n. The stationary point of $\Xi(\mathbf{x}, \mathbf{y}^*)$ occurs at

$$x_{n,a}(\mathbf{y}^*) = \frac{\vartheta + \zeta}{2\beta_k^*} \qquad \forall n, a, \tag{6.16}$$

where $\vartheta = \left[\mathcal{R}_{n,a} \cdot \mathcal{R}(\mathtt{L}_{n,a})\right] / \left[B \cdot \mathcal{C}(\mathtt{S}_{n,a})\right]$ and $\zeta = \beta_k^* - \delta_n^*\mathcal{R}_{n,a} - \tau_{n,a}^*$. The stationary point is obtained through $\nabla_{\mathbf{x}}\Xi(\mathbf{x}, \mathbf{y}^*) = 0$. Using (6.15) and (6.16), the dual

function is obtained, which is given by

$$\begin{aligned}\Upsilon(\boldsymbol{\delta}^*,\boldsymbol{\beta}^*,\boldsymbol{\tau}^*) = &-\sum_{n=1}^{N}\sum_{a=1}^{A}\left[\frac{\vartheta+\zeta}{2\beta_k^*}\left(3\vartheta+\zeta\right)\right]\\ &-\frac{\sum_{n=1}^{N}\sum_{a=1}^{A}\left(\mathcal{R}_{n,a}\frac{\vartheta+\zeta}{2\beta_k^*}\right)\cdot\sum_{n=1}^{N}\sum_{a=1}^{A}\left(\mathcal{R}(\mathtt{L}_{n,a})\frac{\vartheta+\zeta}{2\beta_k^*}\right)}{\left[B\cdot\sum_{n=1}^{N}\sum_{a=1}^{A}\left(\mathcal{C}(\mathtt{S}_{n,a})\frac{\vartheta+\zeta}{2\beta_k^*}\right)\right]^2}\\ &-\sum_{n=1}^{N}\delta_n^*B_n^{\max}-\sum_{n=1}^{N}\tau_{n,a}^*. \end{aligned} \tag{6.17}$$

The dual function is a concave function on $\mathcal{X}_p^\sharp$. The canonical dual problem associated with (6.8) can be formulated as

$$\min\left\{\mathcal{P}(\mathbf{x})|\mathcal{X}_p\right\} = \max\left\{\Upsilon(\boldsymbol{\delta}^*,\boldsymbol{\beta}^*,\boldsymbol{\tau}^*)|\mathcal{X}_p^\sharp\right\}. \tag{6.18}$$

Theorem 4. *If $\mathcal{P}(\tilde{\mathbf{x}}) = \Upsilon(\tilde{\mathbf{y}}^*)$ where $\tilde{\mathbf{x}}$ denotes the KKT point of the primal problem and $\tilde{\mathbf{y}}^* = (\tilde{\boldsymbol{\delta}}^*,\tilde{\boldsymbol{\beta}}^*,\tilde{\boldsymbol{\tau}}^*) \in \mathcal{X}_p^\sharp$ denotes the KKT point of the dual function, there exists a perfect duality relationship between the primal problem in* (6.8) *and its canonical dual problem.*

Proof. The proof directly extends from [111]. ■

Theorem 4 shows that the BIP in (6.8) is converted into a continuous space canonical dual problem which is perfectly dual to it. Moreover, the KKT point of the dual problem provides the KKT point of the primal problem.

Theorem 5. *(global optimality conditions): If $\tilde{\mathbf{y}}^* = (\tilde{\boldsymbol{\delta}}^*,\tilde{\boldsymbol{\beta}}^*,\tilde{\boldsymbol{\tau}}^*) \in \mathcal{X}_p^\sharp$, then $\tilde{\mathbf{x}}$ is a global minimizer of $\mathcal{P}(\mathbf{x})$ over $\mathcal{X}_p$ and $\tilde{\mathbf{y}}^*$ is a global maximizer of $\Upsilon(\tilde{\boldsymbol{\delta}}^*,\tilde{\boldsymbol{\beta}}^*,\tilde{\boldsymbol{\tau}}^*)$ over $\mathcal{X}_p^\sharp$. Hence, $\mathcal{P}(\tilde{\mathbf{x}}) = \min\left\{\mathcal{P}(\mathbf{x})|\mathcal{X}_p\right\} = \max\left\{\Upsilon(\boldsymbol{\delta}^*,\boldsymbol{\beta}^*,\boldsymbol{\tau}^*)|\mathcal{X}_p^\sharp\right\} = \Upsilon(\tilde{\boldsymbol{\delta}}^*,\tilde{\boldsymbol{\beta}}^*,\tilde{\boldsymbol{\tau}}^*)$.*

Proof. The proof directly extends from [111]. ■

According to Theorem 5, if the given global optimality conditions are met, the solution of the canonical dual problem provides an optimal solution to the

primal problem. Solving the KKT conditions associated with the dual function in (6.17) is necessary and sufficient for global optimality as the dual problem is a concave maximization problem over $\mathcal{X}_p^\sharp$.

The KKT conditions of the dual function in (6.17) are given by $(\partial\Upsilon/\partial\delta_n^*) = 0, (\partial\Upsilon/\partial\beta_k^*) = 0$ and $(\partial\Upsilon/\partial\tau_{n,a}^*) = 0$. (6.19), (6.20) and (6.21) give the respective partial derivatives.

$$\begin{aligned}\frac{\partial\Upsilon}{\partial\delta_n^*} =& \sum_{n=1}^{N}\sum_{a=1}^{A}\left[\frac{\mathcal{R}_{n,a}}{4\beta_k^*}\left(3\vartheta+\zeta\right)+3\mathcal{R}_{n,a}\frac{\vartheta+\zeta}{4\beta_k^*}\right] \\ &- \frac{\sum_{n=1}^{N}\sum_{a=1}^{A}\frac{-\mathcal{R}_{n,a}^2}{2\beta_k^*}\cdot\sum_{n=1}^{N}\sum_{a=1}^{A}\left(\mathcal{R}(\mathtt{L}_{n,a})\frac{\vartheta+\zeta}{2\beta_k^*}\right)}{B\cdot\sum_{n=1}^{N}\sum_{a=1}^{A}\left(\mathcal{C}(\mathtt{S}_{n,a})\frac{\vartheta+\zeta}{2\beta_k^*}\right)} \\ &- \frac{\sum_{n=1}^{N}\sum_{a=1}^{A}\left(\frac{-\mathcal{R}(\mathtt{L}_{n,a})\mathcal{R}_{n,a}}{2\beta_k^*}\right)\cdot\sum_{n=1}^{N}\sum_{a=1}^{A}\left(\mathcal{R}_{n,a}\frac{\vartheta+\zeta}{2\beta_k^*}\right)}{B\cdot\sum_{n=1}^{N}\sum_{a=1}^{A}\left(\mathcal{C}(\mathtt{S}_{n,a})\frac{\vartheta+\zeta}{2\beta_k^*}\right)} \\ &+ \frac{\sum_{n=1}^{N}\sum_{a=1}^{A}\left(\frac{B\cdot\mathcal{R}_{n,a}\cdot\mathcal{C}(\mathtt{S}_{n,a})}{2\beta_k^*}\right)\cdot\sum_{n=1}^{N}\sum_{a=1}^{A}\left(\mathcal{R}_{n,a}\frac{\vartheta+\zeta}{2\beta_k^*}\right)}{B\cdot\sum_{n=1}^{N}\sum_{a=1}^{A}\left(\mathcal{C}(\mathtt{S}_{n,a})\frac{\vartheta+\zeta}{2\beta_k^*}\right)} \\ &\times\frac{\sum_{n=1}^{N}\sum_{a=1}^{A}\left(\mathcal{R}(\mathtt{L}_{n,a})\frac{\vartheta+\zeta}{2\beta_k^*}\right)}{B\cdot\sum_{n=1}^{N}\sum_{a=1}^{A}\left(\mathcal{C}(\mathtt{S}_{n,a})\frac{\vartheta+\zeta}{2\beta_k^*}\right)}-\sum_{n=1}^{N}B_n^{\max}, \end{aligned} \tag{6.19}$$

$$\begin{aligned}\frac{\partial \Upsilon}{\partial \beta_k^*} =& -\sum_{n=1}^{N}\sum_{a=1}^{A}\left[\left(\frac{\beta_k^*-\vartheta-\zeta}{4{\beta_k^*}^2}\right)(3\vartheta+\zeta)-\frac{3\vartheta+3\zeta}{4\beta_k^*}\right]\\ &-\frac{\sum_{n=1}^{N}\sum_{a=1}^{A}\left(\mathcal{R}_{n,a}\frac{\beta_k^*-\vartheta-\zeta}{2{\beta_k^*}^2}\right)\cdot\sum_{n=1}^{N}\sum_{a=1}^{A}\left(\mathcal{R}(\mathtt{L}_{n,a})\frac{\vartheta+\zeta}{2\beta_k^*}\right)}{B\cdot\sum_{n=1}^{N}\sum_{a=1}^{A}\left(\mathcal{C}(\mathtt{S}_{n,a})\frac{\vartheta+\zeta}{2\beta_k^*}\right)}\\ &+\frac{\sum_{n=1}^{N}\sum_{a=1}^{A}\left(B\cdot\mathcal{C}(\mathtt{S}_{n,a})\frac{\beta_k^*-\vartheta-\zeta}{2{\beta_k^*}^2}\right)\cdot\sum_{n=1}^{N}\sum_{a=1}^{A}\left(\mathcal{R}_{n,a}\frac{\vartheta+\zeta}{2\beta_k^*}\right)}{B\cdot\sum_{n=1}^{N}\sum_{a=1}^{A}\left(\mathcal{C}(\mathtt{S}_{n,a})\frac{\vartheta+\zeta}{2\beta_k^*}\right)}\\ &\times\frac{\sum_{n=1}^{N}\sum_{a=1}^{A}\left(\mathcal{R}(\mathtt{L}_{n,a})\frac{\vartheta+\zeta}{2\beta_k^*}\right)}{B\cdot\sum_{n=1}^{N}\sum_{a=1}^{A}\left(\mathcal{C}(\mathtt{S}_{n,a})\frac{\vartheta+\zeta}{2\beta_k^*}\right)}\\ &+\frac{\sum_{n=1}^{N}\sum_{a=1}^{A}\left(\mathcal{R}(\mathtt{L}_{n,a})\frac{\beta_k^*-\vartheta-\zeta}{2{\beta_k^*}^2}\right)\sum_{n=1}^{N}\sum_{a=1}^{A}\left(\mathcal{R}_{n,a}\frac{\vartheta+\zeta}{2\beta_k^*}\right)}{B\cdot\sum_{n=1}^{N}\sum_{a=1}^{A}\left(\mathcal{C}(\mathtt{S}_{n,a})\frac{\vartheta+\zeta}{2\beta_k^*}\right)},\end{aligned}\tag{6.20}$$

$$\begin{aligned}\frac{\partial \Upsilon}{\partial \tau_{n,a}^*} =& \sum_{n=1}^{N}\sum_{a=1}^{A}\left[\frac{1}{4\beta_k^*}(3\vartheta+\zeta)+\frac{3\vartheta+3\zeta}{4\beta_k^*}\right]\\ &-\frac{\sum_{n=1}^{N}\sum_{a=1}^{A}\frac{-\mathcal{R}_{n,a}}{2\beta_k^*}\cdot\sum_{n=1}^{N}\sum_{a=1}^{A}\left(\mathcal{R}(\mathtt{L}_{n,a})\frac{\vartheta+\zeta}{2\beta_k^*}\right)}{B\cdot\sum_{n=1}^{N}\sum_{a=1}^{A}\left(\mathcal{C}(\mathtt{S}_{n,a})\frac{\vartheta+\zeta}{2\beta_k^*}\right)}\\ &-\frac{\sum_{n=1}^{N}\sum_{a=1}^{A}\left(\frac{-\mathcal{R}(\mathtt{L}_{n,a})}{2\beta_k^*}\right)\cdot\sum_{n=1}^{N}\sum_{a=1}^{A}\left(\mathcal{R}_{n,a}\frac{\vartheta+\zeta}{2\beta_k^*}\right)}{B\cdot\sum_{n=1}^{N}\sum_{a=1}^{A}\left(\mathcal{C}(\mathtt{S}_{n,a})\frac{\vartheta+\zeta}{2\beta_k^*}\right)}\\ &+\frac{\sum_{n=1}^{N}\sum_{a=1}^{A}\left(\frac{B\cdot\mathcal{C}(\mathtt{S}_{n,a})}{2\beta_k^*}\right)\cdot\sum_{n=1}^{N}\sum_{a=1}^{A}\left(\mathcal{R}_{n,a}\frac{\vartheta+\zeta}{2\beta_k^*}\right)}{B\cdot\sum_{n=1}^{N}\sum_{a=1}^{A}\left(\mathcal{C}(\mathtt{S}_{n,a})\frac{\vartheta+\zeta}{2\beta_k^*}\right)}\\ &\times\frac{\sum_{n=1}^{N}\sum_{a=1}^{A}\left(\mathcal{R}(\mathtt{L}_{n,a})\frac{\vartheta+\zeta}{2\beta_k^*}\right)}{B\cdot\sum_{n=1}^{N}\sum_{a=1}^{A}\left(\mathcal{C}(\mathtt{S}_{n,a})\frac{\vartheta+\zeta}{2\beta_k^*}\right)}-N.\end{aligned}\tag{6.21}$$

As in Chapter 5, an IWO [27] algorithm is deployed for solving the complex non-linear equations associated with the KKT conditions [112].

6.7 Simulation Results

In this section, a cached-enabled cloud-based operator network consisting of four clusters is considered. The performance of our caching schemes is evaluated in terms of return on investment, offloaded traffic, quality metric and cache size, which represent the gain achieved from the viewpoint of the CP, MNO, end-user and MNO, respectively.

As in [116, 117], it is assumed that the video popularity is Zipf-like with a parameter of 0.65 and the video file sizes follow a Pareto (0.25) distribution with a minimum size of 60 megabytes. Without loss of generality, it is assumed caching is performed at the level of entire video objects as in [135]. In order to enable caching at the chunk level, index i can be simply adjusted to represent the i^{th} chunk rather than a video object.

The proposed approach is compared with the hit rate optimal caching algorithm LFU, which caches the most popular video contents [118, 135]. In contrast with the other widely used caching algorithm, LRU, LFU focuses on historical popularity over a long period of time. As a caching technique, our approach also considers a long term content popularity. Therefore, it is pertinent to compare the proposed scheme with LFU. Additionally, the results in [135] confirm the relative loss in hit rate of LRU compared with LFU observed for video on demand (VOD) content.

Three scenarios are considered and the aforementioned metrics are measured in each scenario. In *Scenario 1*, the total number of popular contents varies in the range [500,10000] (with 4 quality layers) whereas the sum fronthaul capacity is set to 25 Gbps. The default number of content is set as 4000 in scenarios 2 and 3. The overall fronthaul capacity is varied in the commercially available range of 15 to 40 Gbps (increments of 5 Gbps) in the former and the total cost from 0 to 1 in the latter. The caching budget constraint is relaxed in scenarios 1 and 2. A comparison of the performance of different caching techniques under the three

TABLE 6.2: Performance Comparison of Caching Techniques

Metric[1]	**Scenario**	**Figure**	**BPA**[2]	MRI (%)	MOT (%)	LFU (%)
ROI	1	Fig. 6.2a	MRI	-	+37.1	+33.7
	2	Fig. 6.3a	MRI	-	+38.2	+32.13
	3	Fig. 6.4a	MRI	-	+16.44	+30.87
CS	1	Fig. 6.2b	MRI	-	−42.27	−21.82
	2	Fig. 6.3b	MRI	-	−34.41	−17.06
	3	Fig. 6.4b	MRI	-	−16.22	−13.51
OT	1	Fig. 6.2c	MOT	+28.45	-	+34.72
	2	Fig. 6.3c	MOT	+25.7	-	+34.5
	3	Fig. 6.4c	MOT	+16.2	-	+28.9
QM	1	Fig. 6.2d	MRI	-	+13.01	+21.82
	2	Fig. 6.3d	MRI	-	+11.74	+21.61
	3	Fig. 6.4d	MRI	-	+12.67	+20.87

[1]ROI: return on investment; CS: cache size; OT: offloaded traffic; QM: quality metric.

[2]BPA: best performing algorithm.

scenarios is illustrated in TABLE 6.2.

In the aforementioned scenarios, the KKT conditions are solved for each dual variable using the IWO algorithm (implemented in MATLAB) and compute the allocation vector $\mathbf{x}_n$ using (6.16). A pseudo code for the resource allocation algorithm is given as Algorithm 4. TABLE 5.1 provides a summary of the simulation parameters for IWO.

6.7.1 Scenario 1 - Variable Content Population

Fig. 6.2 demonstrates the performance of the caching algorithms under *Scenarios 1*. As shown in Fig. 6.2a, for MRI, a growth in the size of the database initially

Algorithm 4: C-RAN caching based on IWO (adapted from [112])

initialize $\boldsymbol{\delta}^*, \boldsymbol{\beta}^*, \boldsymbol{\tau}^*, \forall n \in \mathcal{N}, iter = 0$;

$\forall\ \partial\Upsilon/\partial\boldsymbol{\nu}^*$, where $\boldsymbol{\nu}^* \in (\boldsymbol{\delta}^*, \boldsymbol{\beta}^*, \boldsymbol{\tau}^*)$

create randomly dispersed initial population of Q individuals (weeds):

$\mathcal{W} = \{W_1, \cdots, W_Q\}$;

while $|\boldsymbol{\nu}^*| > \varrho$ *or* $iter = iter_{max}$ **do**

evaluate the fitness of each individual i.e., calculate $f(W_n), \forall n \in \mathcal{W}$

and the colony's best (f_{best}) and worst (f_{worst}) fitness;

sort $\mathcal{W}$ in ascending order according to $f(W_n)$;

select the first Q_p individuals of $\mathcal{W}$ to create the set $\mathcal{W}_p$;

Reproduction:

$\forall\ W_j, j = 1, \cdots, Q_p$

generate

$S_j = \frac{f(W_j) - f_{\text{worst}}}{f_{\text{best}} - f_{\text{worst}}} \times (S_{\max} - S_{\min}) + S_{\max}$ seeds;

create the population of the generated seeds, $\mathcal{W}_s = \{W_s\}$;

Spatial Dispersion:

for $i = 1 : |\mathcal{W}_s|$ **do**

$W_s^i \leftarrow W_s^i + \phi^i$, where $\phi^i \sim L(0, \sigma_{\text{iter}})$;

end

Competitive Exclusion:

create parents and seeds, $\mathcal{W}^* = \mathcal{W} \cup \mathcal{W}_s$;

sort $\mathcal{W}^*$ in ascending order according to fitness;

select the first $Q_{\max}$ individuals of $\mathcal{W}^*$ and create $\mathcal{W}$;

end

select the best fitted individuals $\boldsymbol{\delta}^*, \boldsymbol{\beta}^*$ and $\boldsymbol{\tau}^*$; calculate $\mathbf{x}_n$ using (6.16);

decreases the return on investment. However, once the content population reaches a certain size (≥ 6000), it enters a steady state and remains unchanged. Before reaching a steady state, both overall cache size and offloaded traffic have an increasing behavior as the number of contents rises (see Fig. 6.2b and Fig. 6.2c). However, the growth in the cache size incurs a higher cost than the benefit gained from the increase in the offloaded traffic load, which leads to a gradual decrease in the return on investment. At the point that the return on investment reaches a steady state, the same occurs to cache size and offloaded traffic. This can be justified by the direct relationship between return on investment and the ratio of the return function (related to offloaded traffic) and the cost function (related to cache size).

As can be seen from Fig. 6.2d, there is a slight positive correlation between return on investment and the quality metric. As the content population increases, MRI demonstrates a tendency to cache more video objects in order to prevent the quality metric to be reduced significantly. This in turn decreases the return on investment due to the noticeable rise in cache storage cost.

Likewise, Fig. 6.2a indicates that in case of MOT and LFU, return on investment decreases alongside the increase in the size of the content database. However, due to the cost unawareness nature of the aforementioned schemes, they demonstrate a considerably higher decrease in return on investment in comparison with MRI. As shown in Fig. 6.2c and Fig. 6.2b, by considering both the size and popularity of video contents and not taking storage into account, MOT induces the highest increase in offloaded traffic, and consequently cache storage requirements.

With SVC, in oder to decode a higher video quality representation, all the lower quality layers are needed. Therefore, low video quality layers which are smaller in size and bit rate have greater popularity than high quality layers. Since LFU only takes popularity into consideration, it caches highly popular

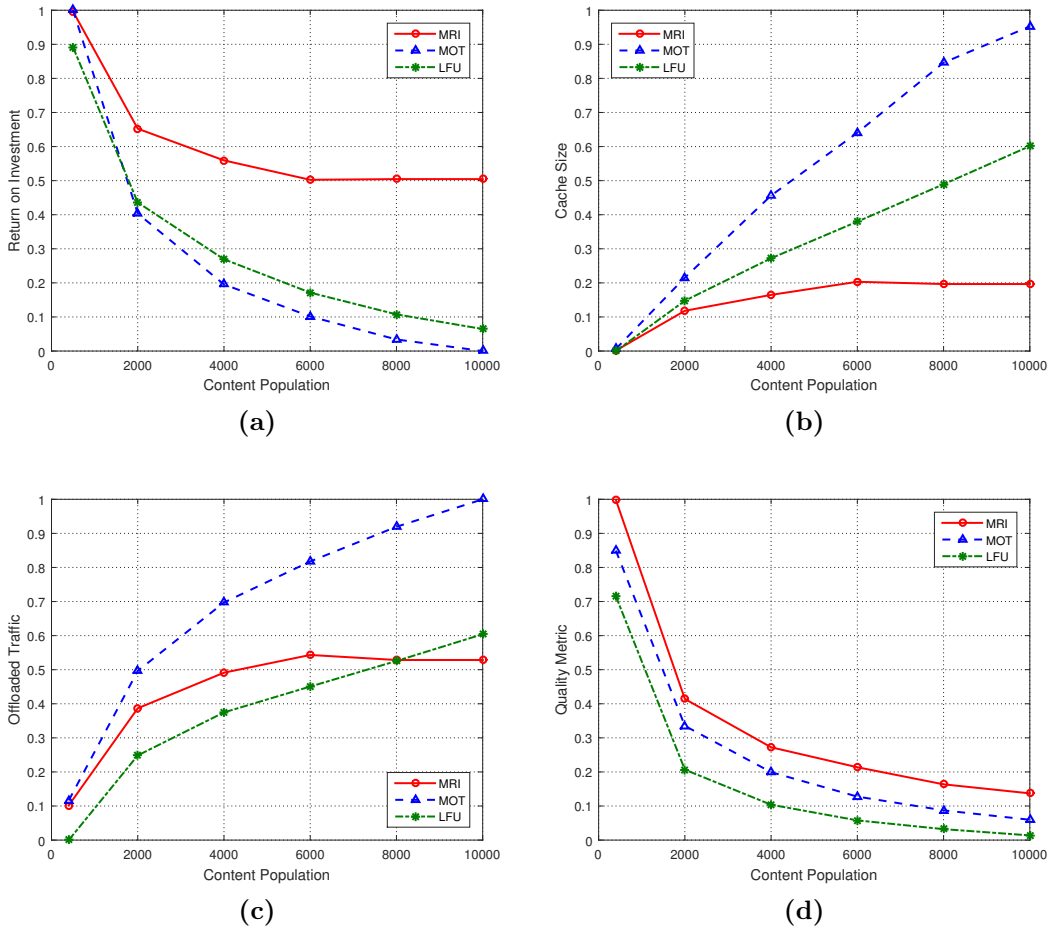


Fig. 6.2: Scenario 1 - varying number of contents: (a) return on investment; (b) cache size; (c) offloaded traffic; (d) quality metric.

videos, which are normally smaller in size and bit rate in comparison with higher quality representations. Hence, as shown in Fig. 6.2b, it leads to lower storage requirements compared with MOT in addition to lower offloaded traffic (Fig. 6.2c) and quality (Fig. 6.2d).

6.7.2 Scenario 2 - Variable Fronthaul Capacity

Fig. 6.3 evaluates the performance of the caching schemes under *Scenarios 2*. Similar to *Scenario 1*, MRI outperforms both MOT and LFU with regard to return on investment, storage efficiency and quality. Likewise, the best performance in terms of increasing the offloaded traffic is achieved by MOT. As the fronthaul

capacity increases, MRI also takes higher bit rate video objects into account, which increases the quality metric significantly as shown in Fig. 6.3d. Therefore, with the increase of the fronthaul capacity, at the cost of a slight reduction in the return on investment (Fig. 6.3a) and storage efficiency (Fig. 6.3b), a considerable increase in the quality metric (Fig. 6.3d) and a satisfactory rise in offloaded traffic load (Fig. 6.3c) are achieved.

For MRI and LFU, increasing the fronthaul capacity relaxes the fronthaul capacity constraint, and hence enables caching more video contents. However, giving priority to video objects that are large in size and popularity, MOT leads to a higher increase in offloaded traffic compared with both MRI, which takes cost into consideration by maximizing the return on investment and LFU, which only considers popularity.

6.7.3 Scenario 3 - Variable Cost

Fig. 6.3 analyzes the performance of the caching schemes under *Scenarios 3*. Unlike the first and second scenarios where no caching budget constraint is set, here we consider maximum budget as the varying factor. Since MRI does not have a budget constraint [see (6.5)], varying the cache budget causes no change to its performance, and hence it demonstrates a static behavior. Similar to *Scenarios 1* and *2*, in this scenario, MRI has a better performance in terms of return on investment, storage efficiency and quality. Likewise, MOT results in a higher increase in offloaded traffic load.

For a fixed number of contents (4000), as the budget constraint increases, MOT continues to cache more contents. As shown in Fig. 6.3c, giving priority to video objects which are large in size and popularity, MOT leads to a higher increase in offloaded traffic in comparison with MRI, which takes cost into consideration by maximizing the return on investment and LFU, which only considers popularity. This in turn causes MOT and MRI to have the lowest and highest

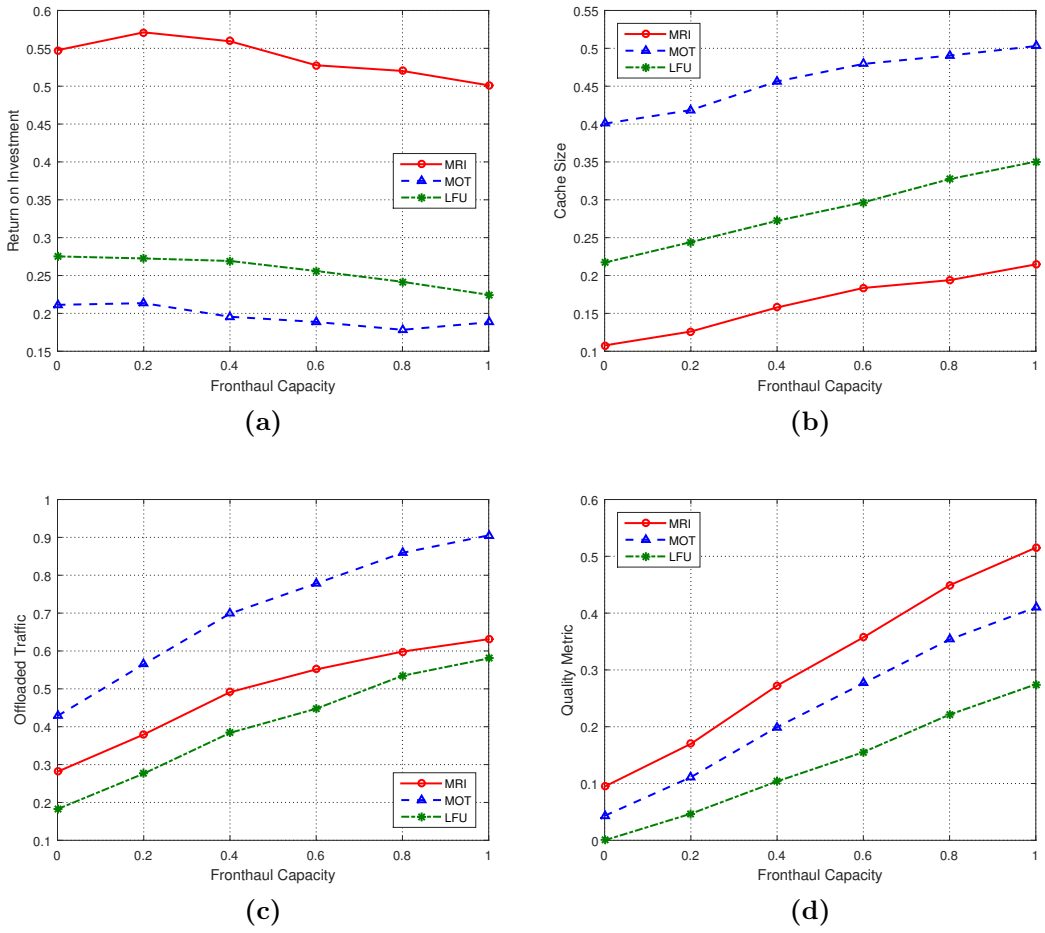


Fig. 6.3: Scenario 2 - varying fronthaul capacity: (a) return on investment; (b) cache size; (c) offloaded traffic; (d) quality metric.

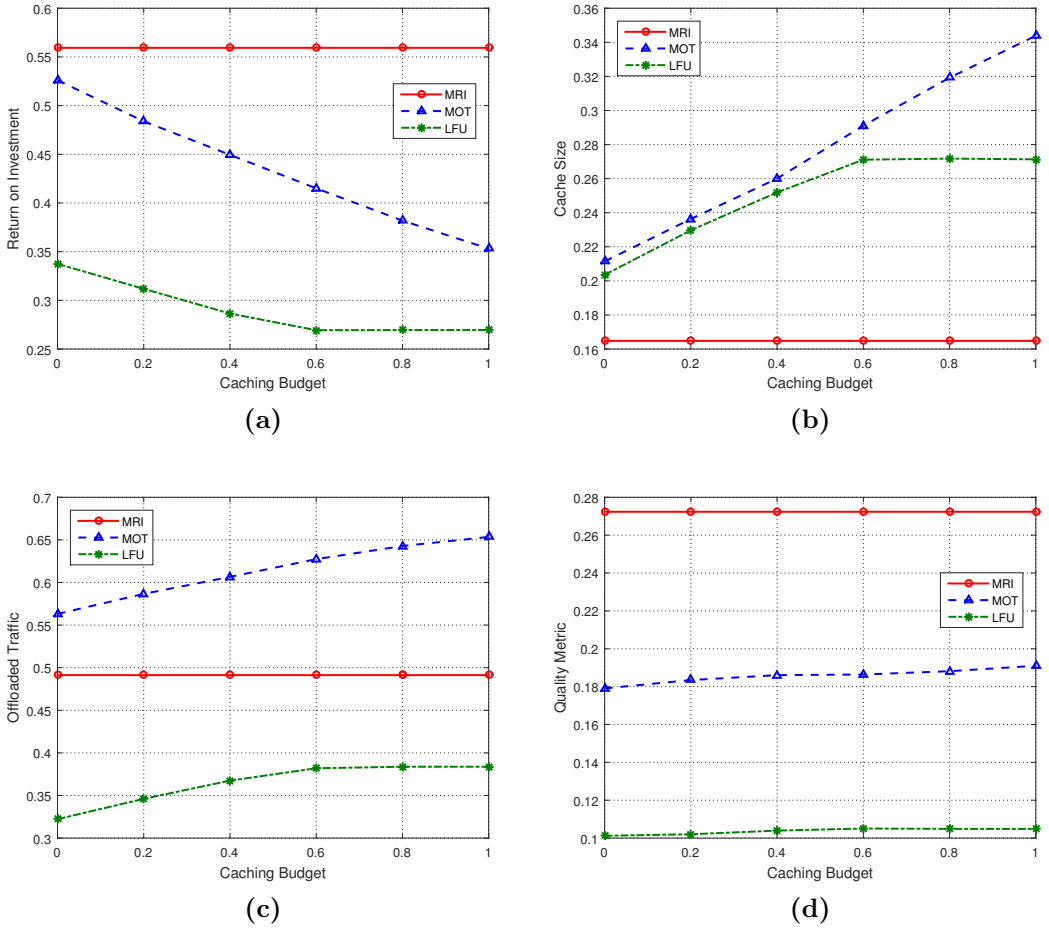


Fig. 6.4: Scenario 3 - varying caching budget: (a) return on investment; (b) cache size; (c) offloaded traffic; (d) quality metric.

storage efficiency, respectively (see Fig. 6.3b).

that after a certain increase in the budget, LFU reaches a steady state as it has already cached the contents with the highest popularity. Caching more contents requires a higher fronthaul capacity, which is set to 25 Gbps in this scenario. However, since MOT takes both the popularity and the size of the objects into consideration, further increase in the budget results in availability of more storage. This leads MOT to cache larger contents (consequently lower storage efficiency). However, it achieves a considerable gain in offloaded traffic. Having cached higher quality representations, MOT exhibits a better performance in terms of quality when compared to LFU.

TABLE 6.3: Average Performance Comparison of Caching Techniques

Metric[1]	**BPA**[2]	MRI (%)	MOT (%)	LFU (%)
ROI	MRI	-	+30.58	+32.23
CS	MRI	-	−31.63	−17.46
OT	MOT	+23.45	-	+32.7
QM	MRI	-	+12.47	+21.43

[1]ROI: return on investment; CS: cache size; OT: offloaded traffic; QM: quality metric.

[2]BPA: best performing algorithm.

6.7.4 Summary

TABLE 6.3 presents a comparison of the average performance of the three caching algorithms under all scenarios. In summary, MRI outperforms the other schemes in terms of return on investment, cache storage efficiency and quality. In comparison with MOT and LFU, MRI leads to an average improvement of 30.58% and 32.23% in return on investment, 31.63% and 17.46% in storage efficiency and 12.47% and 21.43% in quality, respectively. On the other hand, MOT has the best performance with regard to the increase in overall offloaded traffic. It outperforms MRI by 23.45% and LFU by 32.7% .

6.7.5 Complexity Analysis

IWO is an iterative algorithm and is used for each dual variable associated with the dual function in (6.17). In each iteration for $\boldsymbol{\delta}^* \geq 0, \boldsymbol{\beta}^* \geq 0, \boldsymbol{\tau}^* \geq 0$, N, NA and N variables are computed. Therefore, it has an overall worst case complexity of $\mathcal{O}(iter_{\max} \cdot \{2N + NA\})$ [112].

6.8 Conclusion

In this chapter, a CaaS framework for virtual caching in the MNO's infrastructure has been proposed. The first proposed scheme caches video contents in the cloud-based mobile network with the aim of maximizing the return on caching investment. The second approach aims for the maximization of the offloaded traffic as a result of caching, for a given caching budget. CDT is used to convert the proposed BIP virtual caching problem into its canonical dual. Using the IWO algorithm, the solution of the dual problem is obtained. Numerical and simulation results have shown that the proposed framework outperforms LFU algorithm by more than 32%, 21%, 32% and 17% improvements in terms of return on investment, quality, offloaded traffic and storage efficiency, respectively.

Chapter 7

Concluding Remarks and Future Work

7.1 Conclusions

This study was set out to explore different approaches to bring content closer to the end user in order to increase the video capacity of mobile operators' networks and user-perceived video quality, and reduce the end-to-end delay of video streams. One approach to this problem is to cache the highest quality versions of video contents and perform in-network video adaptation. Therefore, two in-network video adaptation schemes have been proposed in order to transrate the video contents cached at the edge of the mobile network.

For the first time, a perceptual quality-aware video adaption scheme which also takes into account statistical delay QoS requirements of video contents has been proposed. This approach adaptively transrates the video in real-time at the mobile edge while guaranteeing a certain level of perceptual quality. It then leads to a statistical delay QoS-aware RA approach, which allocates resources under perceptual quality constraints. This technique results in significant performance enhancement of the proposed system, compared with classical algorithms

in terms of power efficiency, while satisfying the statistical delay-bounded QoS and perceptual quality requirements.

Furthermore, this thesis has presented the first attempt to formulate a video adaptation problem using queuing theory. Hence, a queuing-based QoE-aware in-network video adaptation scheme has been proposed for SVC video streaming in mobile networks, followed by a delay-constrained resource allocation scheme. This technique shows significant performance enhancement in terms of end-to-end delay and power efficiency while satisfying the user's QoE requirements.

This thesis has also followed a different approach by performing in-network proactive caching approaches for caching scalable mobile video contents inside the operator's networks. More specifically, this study has proposed the first cost-effective cache provisioning scheme for hierarchical in-network caching at different nodes in a mobile operator's network.

An answer to the problem of storage provisioning for in-network video caching in an operator's network is provided, which optimizes the trade-off between the cost of transmission bandwidth and the cost of storage. The proposed scheme compared with the widely used LFU algorithm results in significant improvements in return on investment and cost-efficiency.

In light of the trending increase in virtualization of network functions, the first cost-driven CaaS framework for virtual video caching in 5G mobile networks has been presented in this thesis. The proposed CaaS approach maximizes return on caching investment, by finding the best trade-off between the cost of cache storage and bandwidth savings from caching video contents in the MNO's cloud. The CaaS technique has shown significant performance enhancement compared with LFU in terms of return on investment, quality, offloaded traffic and storage efficiency.

Moreover, the following conclusions can be drawn based on the overall picture of the research carried out in this thesis.

- caching video contents in mobile operator networks is a cost-effective solution for operators to serve the increasing demand for mobile video contents in their networks.

- both reactive and proactive video caching increase the video capacity of the wireless network to serve more concurrent videos. However, for reactive video caching, in-network video adaptation and cross-layer resource allocation are required to transrate the video to meet the end-user requirements in terms of user-perceived quality and delay, while minimizing the required resources.

- queuing theory can be used to develop a framework for transrating video streams and performing cross-layer resource allocation.

- compared with the current CDN approach, the in-network caching and CaaS schemes presented in this thesis provide a more cost-effective approach for video streaming over wireless networks. The proposed schemes bring benefits for both the operator and end-user. From the operator's perspective, in-network caching and CaaS reduce the load on the costly backhaul of the operator's network. From the view point of the end-user, they result in reduction in delay and improvement in QoE.

- given the anticipated deployment of cloud infrastructure and virtual network functions in future 5G networks, CaaS can be considered as a cost-effective candidate for streaming video contents over these networks.

This thesis covers some of the key issues in in-network video adaption and caching, and provides the foundations for future research. The following sections highlight a number of challenges that need to be addressed and some research directions for future researchers.

7.2 Future Work

7.2.1 In-Network Video Adaption

The work that has been carried out in this thesis is primarily focused on SVC video coding. An immediate extension to this work is to apply the same framework to high efficiency video coding (HEVC) videos and account for the requirements of HEVC.

The queuing-based in-network video adaptation scheme presented in this work does not take into account the effects of packet loss during transmission over wireless networks. Taking into consideration the effects of transmission packet loss, which further degrade the quality of the transrated video at the receiver would be an interesting extension to the work carried out.

7.2.2 Cross-Layer Resource Allocation

In terms of resource allocation, the main focus in this thesis has been on power minimization. However, a cross-layer resource allocation scheme that maximizes the cumulative data rate and assigns resources to the packets in each temporal/quality layer based on the impact of the layer on the perceived quality of video would be an interesting area for research.

The main focus of this study has been on decreasing queuing delay due to its importance and effects on overall end-to-end delay and jitter. However, future work could look at an optimization problem that also considers other possible delays, such as transmission delay.

As mentioned in Section 7.2.1, future studies should consider the effects of packet loss in transmission on perceived video quality. Therefore, future work would involve cross-layer resource allocation optimization problems which take transmission packet loss into account. An immediate example is to minimize the system power such that video loss tolerance parameters are satisfied while

opportunistic scheduling is performed over fading channels.

7.2.3 Cost-Driven Mobile Video Caching and Caching-as-a-Service

In Chapter 5, the resource limitations of the hierarchical caching levels have not been taken into account. Accounting for computing resources limitations of each caching level would impose an additional constraint to the cache provisioning problem and makes the problem more practical.

Adding a constraint on the budget that the operator is willing to allocate for in-network video caching will also improve the practicality of the problem. Furthermore, it would be interesting to study the practical aspects of CaaS, e.g. service chaining and placement.

This thesis has made the assumption that the cost and return function are linear or logarithmic functions. Deploying other types of functions and performing a sensitivity analysis for the cost and return functions would further clarify the effects of the changes in these functions, and would add further value to the proposed scheme.

In Chapters 5 and 6, the CDT has been used to solve the optimization problems. However, in order to make the scheme more practical, one would define some simplistic heuristics and compare the performance of the proposed IWO approach with the heuristics. Furthermore, comparing the suggested technique with LRU and state of the art caching algorithms would increase the value of this study.

In addition to bandwidth fees, some CDN providers apply other fees such as request and storage fees. Investigating in-network mobile video cache provisioning and CaaS optimization problems that take these fees into consideration and provide a solution that minimizes the bandwidth fees, request fees and storage fees incurred by mobile operators and content providers is a potential area for

future research.

References

[1] Mukaddim Pathan, Rajkumar Buyya, and Athena Vakali. Content delivery networks: State of the art, insights, and imperatives. In *Content Delivery Networks*, pages 3–32. Springer, 2008.

[2] H. Schwarz, D. Marpe, and T. Wiegand. Overview of the scalable video coding extension of the H.264/AVC standard. *IEEE Trans. Circuits Syst. Video Technol.*, 17(9):1103–1120, September 2007.

[3] Kin-Yeung Wong. Web cache replacement policies: a pragmatic approach. *IEEE Netw.*, 20(1):28–34, January 2006.

[4] Cisco. Cisco visual networking index: Global mobile data traffic forecast update, 2015–2020, February 2016.

[5] A. Vakali and G. Pallis. Content delivery networks: status and trends. *IEEE Internet Comput.*, 7(6):68–74, November 2003.

[6] N. Golrezaei, A. Molisch, A. Dimakis, and Caire. Femtocaching and device-to-device collaboration: A new architecture for wireless video distribution. *IEEE Commun. Mag.*, 51:142–149, April 2013.

[7] H. Ahlehagh and S. Dey. Adaptive bit rate capable video caching and scheduling. In *Proc. IEEE Wireless Commun. and Netw. Conf. (WCNC)*, pages 1357–1362, April 2013.

[8] H. Ahlehagh and Shuvashis Dey. Video-aware scheduling and caching in the radio access network. *IEEE/ACM Trans. Netw.*, 22:1444–1462, October 2014.

[9] Guanyu Gao, Weiwen Zhang, Yonggang Wen, Zhi Wang, and Wenwu Zhu. Cost-efficient video transcoding in media cloud by leveraging user viewing pattern. *IEEE Trans. Multimedia*, 17(8):1286–1296, August 2015.

[10] Hatem Abou-Zeid and Hossam Hassanein. Toward green media delivery: Location-aware opportunities and approaches. *IEEE Wireless Commun.*, 21(4):38–46, 2014.

[11] E. Bastug, M. Bennis, and M. Debbah. Living on the edge: The role of proactive caching in 5G wireless networks. *IEEE Commun. Mag.*, 52(8):82–89, August 2014.

[12] H. Ahlehagh and S. Dey. Hierarchical video caching in wireless cloud: Approaches and algorithms. In *Proc. IEEE Int. Conf. Commun. (ICC)*, pages 7082–7087, June 2012.

[13] H. Ahlehagh and S. Dey. Video-aware scheduling and caching in the radio access network. *IEEE/ACM Trans. Netw.*, 22(5):1444–1462, October 2014.

[14] Stefano Spagna, Marco Liebsch, Roberto Baldessari, Saverio Niccolini, Stefan Schmid, Rosario Garroppo, Kazunori Ozawa, and Jun Awano. Design principles of an operator-owned highly distributed content delivery network. *IEEE Commun. Mag.*, 51(4):132–140, 2013.

[15] Xiaofei Wang, Min Chen, Tarik Taleb, Adlen Ksentini, and Victor Leung. Cache in the air: Exploiting content caching and delivery techniques for 5G systems. *IEEE Commun. Mag.*, 52(2):131–139, 2014.

[16] Yao Liu, Fei Li, L. Guo, Bo Shen, and Songqing Chen. A server's perspective of Internet streaming delivery to mobile devices. In *Proc. IEEE Conf. Comput. Commun. (INFOCOM)*, Orlando, USA, March 2012. IEEE.

[17] Felix Hartanto, Jussi Kangasharju, Martin Reisslein, and Keith Ross. Caching video objects: layers vs versions? *Multimedia Tools and Applications*, 31(2):221–245, 2006.

[18] K Chen and R Duan. C-RAN–the road towards green RAN. White Paper, 2011.

[19] T. Taleb, M. Corici, C. Parada, A. Jamakovic, S. Ruffino, G. Karagiannis, and T. Magedanz. EASE: EPC as a service to ease mobile core network deployment over cloud. *IEEE Netw.*, 29(2):78–88, March 2015.

[20] Xiuhua Li, Xiaofei Wang, Chunsheng Zhu, Wei Cai, and V.C.M. Leung. Caching-as-a-service: Virtual caching framework in the cloud-based mobile networks. In *Proc. IEEE Conf. Comput. Commun. Wkshps. (INFOCOM WKSHPS)*, pages 372–377, April 2015.

[21] A. Checko, H.L. Christiansen, Ying Yan, L. Scolari, G. Kardaras, M.S. Berger, and L. Dittmann. Cloud RAN for mobile networks - a technology overview. *IEEE Commun. Surveys Tuts.*, 17(1):405–426, Firstquarter 2015.

[22] H. Hawilo, A. Shami, M. Mirahmadi, and R. Asal. Nfv: state of the art, challenges, and implementation in next generation mobile networks (vEPC). *IEEE Netw.*, 28(6):18–26, November 2014.

[23] Dapeng Wu and R. Negi. Effective capacity: A wireless link model for support of quality of service. *IEEE Trans. Wireless Commun.*, 2(4):630–643, July 2003.

[24] Cheng-Shang Chang. *Performance Guarantees in Communication Networks*. Springer Science & Business Media, 2000.

[25] J. O. Fajardo, I. Taboada, and F. Liberal. Improving content delivery efficiency through multi-layer mobile edge adaptation. *IEEE Netw.*, 29(6):40–46, November 2015.

[26] David Yang Gao. Canonical dual transformation method and generalized triality theory in nonsmooth global optimization. *J. Global Optim.*, 17(1–4):127–160, 2000.

[27] A.R. Mehrabian and C. Lucas. A novel numerical optimization algorithm inspired from weed colonization. *Ecological Informatics*, 1(4):355–366, 2006.

[28] Florin Dobrian, Vyas Sekar, Asad Awan, Ion Stoica, Dilip Joseph, Aditya Ganjam, Jibin Zhan, and Hui Zhang. Understanding the impact of video quality on user engagement. *ACM SIGCOMM Comput. Commun. Rev.*, 41(4):362–373, August 2011.

[29] S.S. Krishnan and R.K. Sitaraman. Video stream quality impacts viewer behavior: Inferring causality using quasi-experimental designs. *IEEE/ACM Trans. Netw.*, 21(6):2001–2014, December 2013.

[30] Iain E Richardson. *The H. 264 advanced video compression standard*. John Wiley & Sons, 2011.

[31] Thomas Schierl, Cornelius Hellge, Shpend Mirta, Karsten Gruneberg, and Thomas Wiegand. Using H. 264/AVC-based scalable video coding (SVC) for real time streaming in wireless IP networks. In *Proc. IEEE Int. Symp. Circuits Syst.*, pages 3455–3458. IEEE, 2007.

[32] Truong Cong Thang, Jae-Gon Kim, Jung Won Kang, and Jeong-Ju Yoo. SVC adaptation: Standard tools and supporting methods. *Signal Process., Image Commun.*, 24(3):214–228, 2009.

[33] ITUT. Recommendation and final draft international standard of joint video specification (ITU-T Rec. H. 264— ISO/IEC 14496-10 AVC). *Joint Video Team (JVT) of ISO/IEC MPEG and ITU-T VCEG, JVTG050*, 33, 2003.

[34] Thomas Wiegand, Gary J Sullivan, Gisle Bjøntegaard, and Ajay Luthra. Overview of the H. 264/AVC video coding standard. *IEEE Trans. Circuits Syst. Video Technol.*, 13(7):560–576, 2003.

[35] Joint Video Team. JSVM software manual. *ITU-T document*, 2008.

[36] Jirka Klaue, Berthold Rathke, and Adam Wolisz. *EvalVid – A Framework for Video Transmission and Quality Evaluation*, pages 255–272. Springer Berlin Heidelberg, Berlin, Heidelberg, 2003.

[37] Jens-Rainer Ohm. Standardization in jvt: Scalable video coding. In *Workshop on Video and Image Coding and Applications (VICA)*, 2005.

[38] T. Schierl, T. Stockhammer, and T. Wiegand. Mobile video transmission using scalable video coding. *IEEE Trans. Circuits Syst. Video Technol.*, 17(9):1204–1217, September 2007.

[39] T. Wiegand, L. Noblet, and F. Rovati. Scalable video coding for IPTV services. *IEEE Trans. Broadcast.*, 55(2):527–538, June 2009.

[40] J. R. Ohm. Advances in scalable video coding. *Proceedings of the IEEE*, 93(1):42–56, January 2005.

[41] P. Amon, T. Rathgen, and D. Singer. File format for scalable video coding. *IEEE Trans. Circuits Syst. Video Technol.*, 17(9):1174–1185, September 2007.

[42] H. Schwarz, D. Marpe, T. Schierl, and T. Wiegand. Combined scalability support for the scalable extension of H.264/AVC. In *Proc. IEEE Int. Conf. Multimedia Expo*, page 4, July 2005.

[43] C. A. Segall and G. J. Sullivan. Spatial scalability within the H.264/AVC scalable video coding extension. *IEEE Trans. Circuits Syst. Video Technol.*, 17(9):1121–1135, September 2007.

[44] M. Kalman, B. Girod, and P. van Beek. Optimized transcoding rate selection and packet scheduling for transmitting multiple video streams over a shared channel. In *Proc. IEEE Int. Conf. Image Process.*, volume 1, September 2005.

[45] Mathias Wien, Renaud Cazoulat, Andreas Graffunder, Andreas Hutter, and Peter Amon. Real-time system for adaptive video streaming based on SVC. *IEEE Trans. Circuits Syst. Video Technol.*, 17(9):1227–1237, 2007.

[46] Oussama Layaida and Daniel Hagimont. Designing self-adaptive multimedia applications through hierarchical reconfiguration. In *Distributed Applications and Interoperable Systems*, pages 95–107. Springer, 2005.

[47] Janet Adams and Gabriel-Miro Muntean. Power save adaptation algorithm for multimedia streaming to mobile devices. In *Proc. IEEE Int. Conf. Portable Inf. Devices*, pages 1–5. IEEE, 2007.

[48] Alisa Devlic. *On Optimization of Quality of User Experience and Wireless Network Bandwidth in Video Content Delivery.* PhD thesis, KTH Royal Institute of Technology, 2015.

[49] P. Juluri, V. Tamarapalli, and D. Medhi. Look-ahead rate adaptation algorithm for DASH under varying network environments. In *Proc. Design Rel. Commun. Netw. (DRCN)*, pages 89–90, March 2015.

[50] Thomas Stockhammer. Dynamic adaptive streaming over HTTP–: standards and design principles. In *Proc. ACM Conf. Multimedia Syst.*, pages 133–144. ACM, 2011.

[51] MG Michalos, SP Kessanidis, and SL Nalmpantis. Dynamic adaptive streaming over HTTP. *J. Eng. Sci. Technol. Rev.*, 5(2):30–34, 2012.

[52] Apostolos Galanopoulos, Georgios Iosifidis, Antonios Argyriou, and Leandros Tassiulas. Green video delivery in LTE-based heterogeneous cellular networks. In *Proc. IEEE Int. Symp. World Wireless, Mobile Multimedia Netw. (WoWMoM)*, pages 1–9. IEEE, 2015.

[53] Yago Sanchez, Thomas Schierl, Cornelius Hellge, Thomas Wiegand, Dohy Hong, Danny De Vleeschauwer, Werner Van Leekwijck, and Yannick Le Louédec. Efficient http-based streaming using scalable video coding. *Signal Process., Image Commun.*, 27(4):329–342, 2012.

[54] C. Müller, D. Renzi, S. Lederer, S. Battista, and C. Timmerer. Using scalable video coding for dynamic adaptive streaming over HTTP in mobile environments. In *Proc. European Signal Process. Conf. (EUSIPCO)*, pages 2208–2212, August 2012.

[55] C. Sieber, T. Hoßfeld, T. Zinner, P. Tran-Gia, and C. Timmerer. Implementation and user-centric comparison of a novel adaptation logic for DASH with SVC. In *IFIP/IEEE Int. Symp. Integr. Netw. Manag.*, pages 1318–1323, May 2013.

[56] S. Ibrahim, A. H. Zahran, and M. H. Ismail. SVC-DASH-M: Scalable video coding dynamic adaptive streaming over HTTP using multiple connections. In *Proc. Int. Conf. Telecommun. (ICT)*, pages 400–404, May 2014.

[57] Yago Sánchez, Cornelius Hellge, Thomas Schierl, Werner Van Leekwijck, Yannick Le Louédec, and France Orange-FT. Scalable video coding based DASH for efficient usage of network resources. In *Position Paper for the Third W3C Web and TV workshop*. Citeseer, 2011.

[58] Jia Wang. A survey of web caching schemes for the Internet. *ACM SIGCOMM Computer Commun. Rev.*, 29(5):36–46, 1999.

[59] Arun Venkataramani, Praveen Yalagandula, Ravindranath Kokku, Sadia Sharif, and Mike Dahlin. The potential costs and benefits of long-term prefetching for content distribution. *Comput. Commun.*, 25(4):367–375, 2002.

[60] Stefan Podlipnig and Laszlo Böszörmenyi. A survey of web cache replacement strategies. *ACM Comput. Surv.*, 35(4):374–398, December 2003.

[61] J. Li, J. Wu, G. Dán, Å Arvidsson, and M. Kihl. Performance analysis of local caching replacement policies for Internet video streaming services. In *Proc. Int. Conf. Softw., Telecommun. Comput. Netw. (SoftCOM)*, pages 341–348, September 2014.

[62] R. Fares, B. Romoser, Z. Zong, M. Nijim, and X. Qin. Performance evaluation of traditional caching policies on a large system with petabytes of data. In *Proc. IEEE Int. Conf. Netw., Archit. and Storage (NAS)*, pages 227–234, June 2012.

[63] Rajeev Motwani and Prabhakar Raghavan. *Randomized Algorithms*. Cambridge University Press, 1995.

[64] S. Jiang and X. Zhang. Making LRU friendly to weak locality workloads: a novel replacement algorithm to improve buffer cache performance. *IEEE Trans. Comput.*, 54(8):939–952, August 2005.

[65] P. Jelenkovic and A. Radovanovic. Asymptotic insensitivity of least-recently-used caching to statistical dependency. In *Proc. IEEE Conf. Comput. Commun. (INFOCOM)*, volume 1, pages 438–447, March 2003.

[66] T. R. Gopalakrishnan Nair and P. Jayarekha. A rank based replacement policy for multimedia server cache using Zipf-like law. *CoRR*, abs/1003.4062, 2010.

[67] Duane Wessels. *Web caching.* " O'Reilly Media, Inc.", 2001.

[68] S. Coleri, M. Ergen, A. Puri, and A. Bahai. Channel estimation techniques based on pilot arrangement in OFDM systems. *IEEE Trans. Broadcast.*, 48(3):223–229, September 2002.

[69] M. Ergen, S. Coleri, and P. Varaiya. QoS aware adaptive resource allocation techniques for fair scheduling in OFDMA based broadband wireless access systems. *IEEE Trans. Broadcast.*, 49(4):362–370, December 2003.

[70] R. Nogueroles, M. Bossert, A. Donder, and V. Zyablov. Performance of a random OFDMA system for mobile communications. In *Proc. Int. Zurich Seminar on Broadband Communications*, pages 37–43, February 1998.

[71] M. Katoozian, K. Navaie, and H. Yanikomeroglu. Utility-based adaptive radio resource allocation in OFDM wireless networks with traffic prioritization. *IEEE Trans. Wireless Commun.*, 8(1):66–71, January 2009.

[72] J. Chuang and N. Sollenberger. Beyond 3G: wideband wireless data access based on OFDM and dynamic packet assignment. *IEEE Commun. Mag.*, 38(7):78–87, July 2000.

[73] R. Laroia, S. Uppala, and Junyi Li. Designing a mobile broadband wireless access network. *IEEE Signal Process. Mag.*, 21(5):20–28, September 2004.

[74] R. Knopp and P. A. Humblet. Information capacity and power control in single-cell multiuser communications. In *Proc. IEEE Int. Conf. Commun. (ICC)*, volume 1, pages 331–335, June 1995.

[75] A.J. Goldsmith and Soon-Ghee Chua. Variable-rate variable-power MQAM for fading channels. *IEEE Trans. Commun.*, 45(10):1218–1230, October 1997.

[76] S. Nanda, K. Balachandran, and S. Kumar. Adaptation techniques in wireless packet data services. *IEEE Commun. Mag.*, 38(1):54–64, January 2000.

[77] A. J. Goldsmith and M. Effros. The capacity region of broadcast channels with intersymbol interference and colored Gaussian noise. *IEEE Trans. Inf. Theory*, 47(1):219–240, January 2001.

[78] Guowang Miao and Guocong Song. *Energy and Spectrum Efficient Wireless Network Design.* Cambridge University Press, 2014.

[79] Patrick Svedman. *Multiuser diversity orthogonal frequency division multiple access systems.* PhD thesis, KTH Signals, Sensors and Systems, 2004.

[80] X. Qin and R. Berry. Exploiting multiuser diversity for medium access control in wireless networks. In *Proc. IEEE Conf. Comput. Commun. (INFOCOM)*, volume 2, pages 1084–1094, March 2003.

[81] Fan Zhang et al. *Quality of Experience-driven Multi-Dimensional Video Adaptation.* PhD thesis, Technische Universität München, 2014.

[82] A.A. Khalek, C. Caramanis, and R.W. Heath. Delay-constrained video transmission: Quality-driven resource allocation and scheduling. *IEEE J. Sel. Topics Signal Process.*, 9(1):60–75, February 2015.

[83] Jia Tang and Xi Zhang. Quality-of-service driven power and rate adaptation over wireless links. *IEEE Trans. Wireless Commun.*, 6(8):3058–3068, August 2007.

[84] Bingquan Li, Shuo Li, Chengwen Xing, Zesong Fei, and Jingming Kuang. A QoE-based OFDM resource allocation scheme for energy efficiency and quality guarantee in multiuser-multiservice system. In *IEEE Global Commun. Conf. Wkshps. (GLOBECOM WKSHPS)*, pages 1293–1297, Anaheim, California, December 2012.

[85] Xiao Xiao, Xiaoming Tao, and Jianhua Lu. QoS-aware energy-efficient radio resource scheduling in multi-user OFDMA systems. *IEEE Commun. Lett.*, 17(1):75–78, 2013.

[86] E. Maani, P. V. Pahalawatta, R. Berry, T. N. Pappas, and A. K. Katsaggelos. Resource allocation for downlink multiuser video transmission over wireless lossy networks. *IEEE Trans. Image Process.*, 17(9):1663–1671, 2008.

[87] Kibeom Seong, M. Mohseni, and J.M. Cioffi. Optimal resource allocation for OFDMA downlink systems. In *Proc. IEEE Int. Symp. Inf. Theory*, July 2006.

[88] Jia Tang and Xi Zhang. Cross-layer-model based adaptive resource allocation for statistical QoS guarantees in mobile wireless networks. *IEEE Trans. Wireless Commun.*, 7(6):2318–2328, June 2008.

[89] Christian Isheden and G.P. Fettweis. Energy-efficient link adaptation with shadow fading. In *Proc IEEE Veh. Technol. Conf. (VTC)*, pages 1–5, Budapest, May 2011.

[90] ATIS Technical Report T1.TR.74. Objective video quality measurement using a peak-signal-to-noise-ratio (PSNR) full reference technique, T1. Technical report, TR, October 2001.

[91] Yen Ou, Zhan Ma, Tao Liu, and Yao Wang. Perceptual quality assessment of video considering both frame rate and quantization artifacts. *IEEE Trans. Circuits Syst. Video Technol.*, 21:286–298, March 2011.

[92] E. Yaacoub, F. Filali, and A. Abu-Dayya. QoE enhancement of SVC video streaming over vehicular networks using cooperative LTE/802.11p communications. *IEEE J. Sel. Topics Signal Process.*, 9(1):37–49, February 2015.

[93] Cheng-Shang Chang. Stability, queue length, and delay of deterministic and stochastic queueing networks. *IEEE Trans. Autom. Control*, 39(5):913–931, May 1994.

[94] Qinghe Du and Xi Zhang. Statistical QoS provisionings for wireless unicast/multicast of multi-layer video streams. *IEEE J. Sel. Areas Commun.*, 28(3):420–433, April 2010.

[95] Wei Yu and R. Lui. Dual methods for nonconvex spectrum optimization of multicarrier systems. *IEEE Trans. Commun.*, 54(7):1310–1322, July 2006.

[96] Stephen Boyd. Ee364b course note, 2015.

[97] A.J. Goldsmith and P.P. Varaiya. Capacity of fading channels with channel side information. *IEEE Trans. Inf. Theory*, 43(6):1986–1992, November 1997.

[98] T.M. Cover and J.A. Thomas. *Elements of Information Theory*. Wiley, 2006.

[99] ETSI TR 125 942 V3.3.0. Universal Mobile Telecommunications System (UMTS); RF system scenarios (3GPP TR 25.942 version 3.3.0 Release 1999). Technical report, ETSI, 06 2002.

[100] T Wiegand, G Sullivan, J Reichel, H Schwarz, and M Wien. Joint scalable video model JSVM-9. *Joint Video Team, Doc. JVT-V202*, 2007.

[101] Charilaos C Zarakovitis, Qiang Ni, Dionysios E Skordoulis, and Marios G Hadjinicolaou. Power-efficient cross-layer design for ofdma systems with heterogeneous QoS, imperfect CSI, and outage considerations. *IEEE Trans. Veh. Technol.*, 61(2):781–798, 2012.

[102] Dimitri P Bertsekas, Robert G Gallager, and Pierre Humblet. *Data networks*, volume 2. Prentice-Hall International New Jersey, 1992.

[103] Maria-Estrella Sousa-Vieira. Suitability of the M/G/∞ process for modeling scalable H.264 video traffic. In *Analytical and Stochastic Modeling Techniques and Applications*, pages 149–158. Springer, 2011.

[104] Z. Wang, E.P. Simoncelli, and A.C. Bovik. Multiscale structural similarity for image quality assessment. In *Asilomar Conf. Signals Syst. Comput.* IEEE, November 2003.

[105] Amin Abdel Khalek, Constantine Caramanis, and RW Heath. A cross-layer design for perceptual optimization of h. 264/SVC with unequal error protection. *IEEE J. Sel. Areas Commun.*, 30(7):1157–1171, 2012.

[106] Haidong Wang and Guizhong Liu. Priority and delay aware packet management framework for real-time video transport over 802.11e WLANs. *Multimedia Tools and Applications*, 69(3):621–641, 2014.

[107] R. Gupta, A. Pulipaka, P. Seeling, L.J. Karam, and M. Reisslein. H.264 coarse grain scalable (CGS) and medium grain scalable (MGS) encoded

video: A trace based traffic and quality evaluation. *IEEE Trans. Broadcast.*, 58(3):428–439, September 2012.

[108] MATLAB User's Guide. The mathworks. *Inc., Natick, MA*, 5:333, 1998.

[109] OPNET Modeler. Opnet technologies inc, 2009.

[110] I.C. Wong, O. Oteri, and W. Mccoy. Optimal resource allocation in uplink SC-FDMA systems. *IEEE Trans. Wireless Commun.*, 8(5):2161–2165, May 2009.

[111] David Yang Gao, Ning Ruan, and Hanif D Sherali. Canonical dual solutions for fixed cost quadratic programs. In *Optimization and optimal control*, pages 139–156. Springer, 2010.

[112] A. Aijaz, M. Tshangini, M.R. Nakhai, Xiaoli Chu, and A.-H. Aghvami. Energy-efficient uplink resource allocation in LTE networks with M2M/H2H co-existence under statistical QoS guarantees. *IEEE Trans. Commun.*, 62(7):2353–2365, July 2014.

[113] David Yang Gao, Ruey lin Sheu, Soon yi Wu, and Kok Lay Teo. Canonical dual approach for solving 0-1 quadratic programming problems. *J. Ind. Manag. Optim.*, 4:125–142, 2007.

[114] A. Ahmad and M. Assaad. Polynomial-complexity optimal resource allocation framework for uplink SC-FDMA systems. In *Proc. IEEE Global Telecoms. Conf. (GLOBECOM)*, pages 1–5, December 2011.

[115] Ebrahim Pourjafari and Hamed Mojallali. Solving nonlinear equations systems with a new approach based on invasive weed optimization algorithm and clustering. *Swarm and Evolutionary Computation*, 4:33–43, 2012.

[116] Phillipa Gill, Martin Arlitt, Zongpeng Li, and Anirban Mahanti. Youtube traffic characterization: A view from the edge. In *ACM SIGCOMM Conf. Internet Meas.*, pages 15–28, New York, NY, USA, October 2007. ACM.

[117] Meeyoung Cha, Haewoon Kwak, P. Rodriguez, Yong-Yeol Ahn, and Sue Moon. Analyzing the video popularity characteristics of large-scale user generated content systems. *IEEE/ACM Trans. Netw.*, 17(5):1357–1370, October 2009.

[118] Jia Wang. A survey of web caching schemes for the internet. *ACM SIGCOMM Comput. Commun. Rev.*, 29(5):36–46, October 1999.

[119] John Ardelius, Björn Grönvall, Lars Westberg, and Ake Arvidsson. On the effects of caching in access aggregation networks. In *Proc. ACM ICN Workshop on Information-centric Networking*, 2012.

[120] A. Gharaibeh, A. Khreishah, B. Ji, and M. Ayyash. A provably efficient online collaborative caching algorithm for multicell-coordinated systems. *IEEE Trans. Mobile Comput.*, PP(99), September 2015.

[121] George Pallis and Athena Vakali. Insight and perspectives for content delivery networks. *Commun. ACM*, 49(1):101–106, January 2006.

[122] Zhe Li and G. Simon. In a Telco-CDN, pushing content makes sense. *IEEE Trans. Netw. and Serv. Manag.*, 10(3):300–311, September 2013.

[123] James Broberg, Rajkumar Buyya, and Zahir Tari. Metacdn: Harnessing 'storage clouds' for high performance content delivery. *J. Netw. Comput. Appl.*, 32(5):1012–1022, 2009.

[124] Menglan Hu, Jun Luo, Yang Wang, and B. Veeravalli. Practical resource provisioning and caching with dynamic resilience for cloud–based content

distribution networks. *IEEE Trans. Parallel Distrib. Syst.*, 25(8):2169–2179, August 2014.

[125] J. Z. Wang, Z. Du, and P. K. Srimani. Network cache model for wireless proxy caching. In *Proc. IEEE Int. Symp. Model. Anal. Simulat. Comput. Telecommun. Syst.*, pages 311–314, September 2005.

[126] W. H. O. Lau, M. Kumar, and Svetha Venkatesh. A cooperative cache architecture in support of caching multimedia objects in MANETs. In *Proc. 5th ACM Int. Wkshp Wireless Mobile Multimedia*, WOWMOM '02, pages 56–63, New York, NY, USA, 2002. ACM.

[127] Dilip Kumar Krishnappa, Samamon Khemmarat, Lixin Gao, and Michael Zink. On the feasibility of prefetching and caching for online tv services: a measurement study on hulu. In *Passive and Active Measurement*, pages 72–80. Springer, 2011.

[128] Liangzhong Yin and Guohong Cao. Supporting cooperative caching in ad hoc networks. *IEEE Trans. Mobile Comput.*, 5(1):77–89, January 2006.

[129] N. Golrezaei, K. Shanmugam, A.G. Dimakis, A.F. Molisch, and G. Caire. FemtoCaching: Wireless video content delivery through distributed caching helpers. In *Proc. IEEE Conf. Comput. Commun. (INFOCOM)*, pages 1107–1115, March 2012.

[130] K. Poularakis, G. Iosifidis, and L. Tassiulas. Approximation algorithms for mobile data caching in small cell networks. *IEEE Trans. Commun.*, 62(10):3665–3677, October 2014.

[131] Jun Li, Youjia Chen, Zihuai Lin, Wen Chen, B. Vucetic, and L. Hanzo. Distributed caching for data dissemination in the downlink of heterogeneous networks. *IEEE Trans. Commun.*, 63(10):3553–3568, October 2015.

[132] Rajkumar Buyya, James Broberg, and Andrzej M Goscinski. *Cloud computing: principles and paradigms*, volume 87. John Wiley & Sons, 2010.

[133] V. Jacobson. Congestion avoidance and control. *ACM SIGCOMM Comput. Commun. Rev.*, 18(4):314–329, August 1988.

[134] Seyed Ehsan Ghoreishi, Dmytro Karamshuk, Vasilis Friderikos, Nishanth Sastry, and A. Hamid Aghvami. Provisioning cost-effective mobile video caching. In *IEEE Int. Conf. Commun. (ICC)*, page to appear, May 2016.

[135] C. Fricker, P. Robert, J. Roberts, and N. Sbihi. Impact of traffic mix on caching performance in a content-centric network. In *Proc. IEEE Conf. Comput. Commun. Wkshps. (INFOCOM WKSHPS)*, pages 310–315, March 2012.